Dependability of IT Systems in Emergency Situations – Theory and Practice

Weyns, Kim

2008

Dependability of IT Systems in Emergency Situations – Theory and Practice

Kim Weyns

Licentiate Thesis, 2008

Department of Computer Science
Lund University
Faculty of Engineering

ISSN 1652-4691
Licentiate Thesis 8, 2008
LU-CS-LIC:2008-1

Department of Computer Science
Faculty of Engineering
Lund University
Box 118
SE-221 00 Lund
Sweden

Email: kim.weyns@cs.lth.se

To Maja

Abstract

As our dependence on IT systems increases, evaluating the dependability of critical IT systems becomes more important. One of the main challenges in software reliability engineering is the sensitivity of software systems to a changing usage. This is especially important for systems that are critical in the aftermath of a crisis and for which reliability is the most important aspect of dependability. The crisis might change the usage of the system, and this could have a negative effect on the reliability. Because crisis situations are typically rare events, both the reliability and the criticality of IT systems after a crisis situation are hard to predict.

The first part of this thesis focuses on the analysis of the sensitivity of the reliability of IT systems to changes in their usage. With the help of statistical methods the effects of changing usage profiles, modelled through the use of Markov models, can be examined. After a theoretical derivation of the properties of different models for the usage of software systems, the results are validated by applying the models to the data collected from the logfiles of a webserver.

Swedish municipalities also depend more and more on IT systems for their daily work. Because of their important role in the relief coordination after a crisis, the dependability of their IT systems during these emergency situations is especially critical. The evaluation of this dependability requires the combination of two kinds of information: how critically needed the IT systems are in the aftermath of a crisis and how trustworthy the critical systems are.

To avoid that a failing IT system disturbs the relief work, risk and vulnerability analyses need to take into account the dependability of critical IT systems. This way, municipalities can make sure that the relief work is not critically dependent on systems that are not sufficiently reliable.

The second part of this thesis describes a case study on how two Swedish municipalities deal with these issues. The study focuses especially on the division of responsibilities in the municipalities and on their current methods. The study shows that today there is much room for improvement, especially in the communication between IT personnel and emergency managers. The main goal of these case studies is to form a basis for the development of practical methods that can assist Swedish municipalities in evaluating the dependability of their IT systems and in integrating this information in their emergency planning in the near future.

Acknowledgements

The work presented in this thesis was funded by the Swedish Emergency Management Agency under grant for FRIVA, Framework Programme for Risk and Vulnerability Analysis of Technological and Social Systems.

There are many people without whom it would have been impossible for me to write this thesis. First of all, I wish to thank my supervisors, Per Runeson and Martin Höst, for giving me the chance to be part of their team, for their guidance during the research and for their encouragements along the way.

Secondly, I wish to thank all my colleagues at the departments of Communication Systems and Computer Science, for giving me a fantastic environment to work in. There is nothing more important as a motivation than having a place you can feel happy to go to work every day.

I of course also want to thank the participants in the interviews and the many researchers I have met throughout the last years with whom I could discuss my ideas, not in the least my colleagues in the FRIVA project. The inspiration from all these discussions contributed to this thesis in so many ways.

I also want to specially thank the Swedish Emergency Management Agency (SEMA) for giving me the chance to work with these challenges while at the same time investing in a more dependable Swedish society.

And last but not least, I would not be where I am today without the continuous support and motivation from Kerstin, Maja, and my family and friends, both in Sweden and in Belgium. I cannot thank you enough!

Contents

Chapter 1

Introduction

IT systems have become an essential part of our society. This evolution has not only created new opportunities, but also new threats to our modern society. The presence of IT systems everywhere has made us dependent on IT systems for our daily life. At the same time as the usage of IT systems increases, the number of reports of problems caused by failures of critical IT systems has also increased (Neumann 2006).

One of the common aspects of these failures is the faith in systems that are not sufficiently dependable. The analysis of the dependence on a certain IT system or a set of IT systems requires the combination of two different activities. Firstly, there is a need for an analysis of how critical the IT system is, or in other words what the consequences would be of the system failing or being unavailable. Secondly, this requires an analysis of how trustworthy the systems are, or more precisely what the chance is that different kinds of failures occur. The two analyses require different techniques. Because they also require different special expertise, they are often done by different people. Each of the analyses is far from easy to do well, and the situation is further complicated by the fact that both the criticality and the reliability of a software system are dependent on how the system is used. This usage typically changes over time and can be hard to predict.

A special case of critical dependence on IT systems is when IT systems are used in emergency management and the recovery efforts after a crisis. Just as in other sectors in our society, IT systems have also become a critical resource in emergency management. In emergency situations, a number of IT systems that are not critical in everyday situations can suddenly become highly critical. This need for a high availability during extreme conditions poses special requirements on the dependability of these IT systems. Typical systems that are not highly critical under normal conditions but that can become critical under crisis conditions are for example telecommunication systems, geographical information systems and demographic information systems.

At the same time the usage of these systems can change drastically and the emergency situations can directly and indirectly influence the reliability of these IT systems. For

example, after a major storm the telecommunication infrastructure might be badly damaged and might be further stressed by an increased number of emergency calls.

This thesis approaches the problem of the evaluation of the dependability of IT systems from two different angles. A first, theoretical part focuses on usage models for software systems that can be used to model and predict the sensitivity of the reliability to changes in their usage. A second, more practical part, focuses on how Swedish municipalities deal on a daily basis with incorporating their IT systems in emergency management.

1 Research Goals

The research presented in this thesis is part of the FRIVA, Framework Programme for Risk and Vulnerability Analysis, research project funded by the Swedish Emergency Management Agency. All of the research within FRIVA concerns society's resilience against events that can affect critical societal functions. The research in this thesis is focussed on the role of IT systems in emergency management, and more specifically on methods for evaluating the dependability of IT systems in such crisis situations.

The goal of the research presented in this thesis was originally formulated as:

Goal 1 *To develop and evaluate methods for the analysis of the dependability of IT and communication systems that also take into account the risk of catastrophic events*

This goal implicitly contained the following important aspects:

1. To study how current methods used in software reliability engineering could be adapted to take into account the risk for catastrophic events
2. To contribute to the research front by developing, improving and evaluating methods for the evaluation of software dependability with special focus on rare, but critical events
3. To study how these methods could be practically used today or in the near future by the main actors in the Swedish emergency management system

The first two of these aspects are of a more theoretical nature and are mostly explored in the first part of this thesis. Because of the central role of Swedish municipalities in the Swedish emergency management system, the second part of this thesis focuses mostly on how municipalities deal with the evaluation of the dependability of their IT systems in crisis situations.

The last of the three goals is especially important because the typical Swedish municipality is different from the organisations that are the traditional target of the latest research in software reliability engineering, such as the telecommunication and avionics sector. Because little was published before about the current methods used at Swedish municipalities today, there was a need for a first explorative study to investigate the current status and current needs of Swedish municipalities concerning the

issues discussed above. Only in a next step methods can be developed that can easily be used by Swedish municipalities today.

These different aspects of the goal of this research can create a conflict between conducting research that can easily be generalized to an international environment and producing results that are easily applicable in a practical Swedish setting. This thesis tries to keep a balance between the two by including some results that are more general but also some results that are more specific for Swedish municipalities.

2 Concepts and Definitions

This thesis uses a large number of concepts that are used in different meanings, both in everyday conversation and in scientific literature. To avoid confusion, their exact meaning in this thesis is clarified shortly in this section.

For all software engineering (Sommerville 2005) concepts concerning *dependability* we will follow the definitions from Avizienis et al. (2004). This means that dependability is defined as the most general concept, encompassing the more limited concepts of reliability, availability, safety and security. Basically, *reliability* is mostly concerned with how often failures occur or with the chance of failure-free operation. *Availability* measures the percentage of time the system is accessible by taking into account how long the system is not functioning when failures occur. *Safety* is concerned with the absence of failures causing catastrophic consequences for its users and the environment, while *security* describes how sensitive the system is to external threats. The more general concept of *dependability* takes into account all these aspects and corresponds best to the intuitive notion of how much a system can safely be depended upon by its users. In the work of Neumann (2006), dependability is referred to as *trustworthiness*, which is then of course the opposite of *untrustworthiness*.

Typical for complex software systems is that the reliability of the systems is highly dependent on how the system is used. Complex software systems typically contain a large amount of code and it is nearly unavoidable that some faults are present in the code (Hatton 1997). The probability that these faults result in failures depend heavily on how and how often the respective part of the code is used by the software's users. To model this dependency on how the software is used, software reliability engineering uses the concept of *usage profiles*. Usage profiles, also sometimes called *operational profiles*, can be used for prioritising testing to the most used components (Musa 1993, Trammell 1995), but also for calculating the reliability of the whole system from the reliability of its components (Siegrist 1988). One way to model these usage profiles in a more formal way is by using Markov chains (Le Guen et al. 2004), a special kind of discrete-time stochastic processes.

Also, a number of concepts from the field of emergency management need to be defined. Risk analysis and vulnerability analysis are often mentioned together, or even used to refer to the same process. In this thesis we will nevertheless try to make a distinction between the two.

Risk analysis is concerned with the identification of possible threats, with the prob-

> "Crises are events that disrupt the functioning of society or jeopardise the conditions that govern the life of the population. They include serious crises in times of peace as well as war. Such situations demand good emergency management if they are not to undermine confidence in the Government and authorities and potentially threaten the national security and democracy of Sweden." - SEMA

Figure 1.1: SEMA's definition of a crisis

ability that these possible incidents occur and with their immediate consequences. *Vulnerability analysis* on the other hand is concerned with the study of how severe the consequences are of possible incidents on society. For example, risk analysis might study what the possibility is of a large flood caused by a dam failure or by extreme rainfall, while vulnerability analysis would study what the consequences would be of a severe flood and its impact on the population.

From this it should be clear that risk and vulnerability analysis often go hand in hand, since the true value lies in the combination of both. In short, one could say that it is important to reduce the risk for those incidents to which society is the most vulnerable and to prioritise reducing the vulnerability to those events for which the risk is the highest. A more elaborate discussion of the concepts risk and vulnerability can for example be found from Johansson (2007).

The Swedish Emergency Management Agency, SEMA, defines a *crisis* as in Figure 1.1. In other words, a crisis is when a combination of events, e.g. natural hazards, accidents or sabotage, result in a situation that negatively affects society in a way that hinders vital society functions. Examples of crises that are included in this definition might be terror attacks, storms, floods, or even a long-term disruption in the national telephone network or the Internet.

Emergency management, also called crisis management is the process of preparing for and dealing with crisis situations. In Sweden emergency management is mainly a task of the local and regional governments, a task in which they are supported by SEMA. Before the crisis occurs, emergency management includes mitigation and preparation measures. The *mitigation* process consists of reducing the probability or consequences of possible crises. *Preparation* is concerned with developing emergency plans. It is clear that risk and vulnerability analysis is an essential part of these two first phases of emergency management.

Also the *emergency response* during a crisis is an essential part of emergency management. During this phase the focus is on coordinating all the emergency response resources to minimize the crisis' effect on society. After a crisis, the part of emergency management concerned with trying to restore the affected society to normal conditions, is called *emergency recovery*.

With these concepts defined, we can now situate the research presented in this thesis more formally within the research fields of software engineering and emergency man-

agement. From a software engineering perspective, this thesis deals with the more general focus of dependability, but with a stronger focus on safety than on security. The first part of the thesis is more strictly focussed on reliability, and more specifically on the effects of changing usage profiles.

From an emergency management point of view, the work in this thesis concerns the risk and vulnerability analyses of IT systems conducted in the mitigation and preparation phase of emergency management. More exactly, it discusses methods used for evaluating the vulnerability caused by depending on possibly untrustworthy IT systems in the response and recovery phases after a crisis.

3 Outline of the Thesis

The work presented in this thesis is divided into two parts. The first part contains theoretical contributions to the field of software reliability engineering and their applications. This first part started from the latest research on reliability models and adapted and extended some of those methods to take into account changing usage profiles.

One of the goals of this research was to also contribute with results that Swedish municipalities can apply in their everyday work. Therefore, there was also a need for research that is much closer to the current state of practice for software dependability. Since little is actually written about the current state of practice at Swedish municipalities, the second part of this thesis focuses on exploring how Swedish municipalities today deal with dependability issues of their IT systems in crisis situation, and which problems they are experiencing. The main goal of this part was to build a stable base for further research on how modern techniques from software reliability engineering could be brought into practice to help Swedish municipalities deal with these issues.

Both parts of this thesis contain complete papers as they were published at respective conferences with only minor editorial changes to create a uniform reading experience throughout this thesis. The papers are presented in the order they were written. The author of this thesis is also first author of all the papers presented in this thesis and is responsible for most of the results presented in each of the papers.

3.1 Sensitivity of Software Reliability

The first, more theoretical, part of this thesis contains three papers about the sensitivity of software reliability estimates, based on usage profiles models, to changes in the underlying usage profile.

The first of the three papers contains the theoretical background where the basic models are derived and the theoretical background is explained in detail. The second of the three papers shortly repeats the theory of the first paper and then describes how more advanced models can be derived and discusses a number of alternative models that are more realistic than the basic model.

Because the results of the second paper are so closely connected to the results of the first, the main results of the first paper are repeated there, which creates some overlap between the two papers. Because of strict limitations on the length of the second paper, the derivation of the results and the examples are described in much greater detail in the first paper. Therefore the first paper is still included in this thesis although the second paper is meant to replace and extend the first.

The first two papers only use a fictive example with a usage model taken from literature. The last of the three papers uses real data to show how the models from the first two papers can be applied to a real system. This paper discusses how webserver reliability can be measured and analysed with the help of these models. Further, this last paper also contains a thorough discussion of the data processing necessary for the extraction of a usage profile from the logs of a webserver.

Part I of this thesis contains the following three papers:

PAPER I:
Sensitivity of System Reliability to Usage Profile Changes
Kim Weyns and Per Runeson
Proceedings of the Fifth Conference on Software Engineering Research and Practice in Sweden (SERPS'05), Västerås, October 2005

PAPER II:
Sensitivity of Software System Reliability to Usage Profile Changes
Kim Weyns and Per Runeson
Proceedings of the 22nd Annual ACM Symposium on Applied Computing, Seoul, Korea, March 2007

PAPER III:
Sensitivity of Website Reliability to Usage Profile Changes
Kim Weyns and Martin Höst
Proceedings of the 18th IEEE International Symposium on Software Reliability Engineering (ISSRE 2007), Trollhättan, Sweden, November 2007

3.2 Dependability at Swedish Municipalities

The second part of this thesis contains one paper describing the results of two case studies at two Swedish municipalities. The case studies explore how these two municipalities incorporate their IT systems in the risk and vulnerability analyses they conduct to assess the dependability of their IT systems in emergency situations. Special focus was given to the division of responsibilities for these issues, to the methods being used and to the problems experienced today.

The research described in this part should be seen as the first step in a larger research project on the same topic. Since one of the initial goals of this research was to develop methods that could be practically used by municipalities, it was important to understand the target group's current situation and needs. Many interesting software

engineering methods have been developed in the last years that could help an organisation deal with the dependability of their IT systems, but most of them are unsuitable for Swedish municipalities at this time. This can, for example, be because they are too advanced, because they are more directed at large organisations or because they do not take into account the special operative role of municipalities in the Swedish emergency management system. To be able to focus the next steps in the research, it was first necessary to understand the main problems that municipalities are facing today, and this is exactly what part II of this thesis focuses on. The planned next steps for this research are described in more detail in section 7.

Part II of this thesis contains the following paper:

> **PAPER IV:**
> **Dependability of IT Systems in Emergency Management at Swedish Municipalities**
> Kim Weyns and Martin Höst
> *Seventh Conference on Software Engineering Research and Practice in Sweden (SERPS'07), Göteborg*, October 2007

An extended version of this paper, including more details on future work, will be submitted to an international conference shortly after the publication of this thesis. The planning and motivation for the case studies presented in the paper above, were previously presented in the following extended abstract, not included in this thesis:

> **EXTENDED ABSTRACT:**
> **Software Dependability under Emergency Conditions**
> Kim Weyns and Per Runeson
> *Presented at The 17th IEEE International Symposium on Software Reliability Engineering (ISSRE 2006), Government Track, Raleigh, North Carolina, USA*, November 2006

4 Related Work

The first, more theoretical part of this thesis builds directly on the earlier software reliability research on Markov models done by Siegrist (1988) and Goševa-Popstojanova and Kamavaram (2004). The concept of Markov chains has been used extensively in software reliability engineering. For example, Le Guen et al. (2004) use Markov models to generate test cases and to combine the results of the testing to a total reliability estimate. Markov models can also be used to calculate test coverage, as done by Walton and Poore (2000). The sensitivity to usage profile changes in test coverage has been discussed by Wesslén et al. (2000).

Much less related work is available for the case study presented in part II of this thesis, especially in academic research. SEMA has published many reports and recommendations about risk and vulnerability analysis in Swedish emergency management, such

as the work by Sundelius et al. (1997) and Hallin et al. (2004), but not so many deal with the role of IT systems. SEMA's publications such as BITS (2003) and the first results of the survey discussed in the case studies (Kalmelid and Gustavsson 2005) focus often more on security than on reliability of IT systems. An important collection of related work to the results in this part of the thesis consists of reports of famous failures of IT systems in crisis situations, such as those reported by National Research Council (2003), Rahman et al. (2006) or the Swedish National Post and Telecom Agency (PTS 2005).

More related work that is specific to one of the two parts of this thesis, can be found in the respective articles. One example of work that is directly related to the research in both parts of this thesis concerns the usage of Markov models for doing risk and vulnerability analysis of IT systems at the architectural level (Yacoub and Ammar 2002). They combine Markov models and risk and vulnerability analysis, but the main difference with the work in this thesis is that their analysis method is designed for small, specialised systems, and is less suited for complex systems that are part of a municipality's emergency management.

An area of software reliability engineering that is closely related to both parts of this thesis is the evaluation of the reliability of IT systems with very high reliability. A number of different approaches have been described that try to tackle this problem. Chan (2004) describes the advantages and disadvantages of accelerated stress testing and how it can be used to discover faults or weaknesses in the systems that normal usage testing would probably not detect. Also Tang and Hecht (1997) and Voas and Miller (1995) discuss the importance of testing outside the normal usage profile to discover rare failure modes of the system. Thomason and Whittaker (1999) discusses the first steps to how even rare failures can be modelled with Markov chains.

Also, different techniques have been developed for risk and vulnerability analysis of critical IT systems. For example, Rae et al. (2005) describe a technique for conducting dependency analysis called *critical feature analysis*. Also Lawrence and Gallagher (1997) discuss in detail how a software safety hazard analysis can be conducted. Both these papers stress the importance of looking systematically at all possible failure modes. These methods are well suited for the analysis of technical systems containing a large software component, but a higher level analysis is needed when the IT systems are part of a large emergency management organisation. Nevertheless a municipal risk and vulnerability analysis could contain such a detailed technical analysis of the most critical IT systems.

Finally, a number of authors (Hatton 1997, Fenton and Pfleeger 1998) have discussed the general difficulty of constructing and verifying highly reliable systems. They state that we need to learn from past failures and should be more cautious before relying on systems that have not been shown to be reliable, because we currently have no easy way to build highly complex software systems that are guaranteed to be highly dependable.

5 Research Methodology

This thesis does not only contain the results of the research work but also discusses the research methodology in detail. The research methodology is the most important factor in assuring the validity of the results. Just as the research goal for the two parts of this thesis is different, the methodology used is substantially different.

Part I of this thesis is based on the mathematical derivation of statistical properties of Markov models used in software reliability modelling. To assure the validity of the results, the theoretical results are compared to corresponding values obtained from extensive Monte-Carlo simulations on small fictive examples from the literature. To investigate the applicability of the results, the theory is finally also applied to a larger real-life system.

This methodology was chosen because it connected well to the work done before on Markov models, and because it allowed for a theoretical study that could be started quickly without first setting up contacts for empirical research. Because of the difficulty of obtaining sufficiently detailed usage profiles from real life systems, the first part of the research employed only fictive examples. Only when more research was done on how a detailed usage profile could be derived from the log files of webserver, the results could be validated on a real-life system.

Part II of this thesis, on the other hand, contains mainly empirical research. The methodology described in this part of the thesis is mainly based on the work by Robson (2002) and Yin (2003). This part describes two case studies that were mainly based on a series of interviews and a number of collected documents, complemented with data from a large survey. Also for this research the validity is the main concern. Therefore, measures to reduce the effects of threats to the validity, such as researcher bias, were taken throughout the course of this research.

The practical aspect of the research questions in part II required a substantially different approach from the research in Part I, and the questions could only be answered through empirical research. The methodology for this research was selected from empirical methods commonly used for empirical software engineering research (Shull et al. 2007). These methods allow for a high traceability of the results and reduce the threats to the validity.

More details on the research methodology, the threats to the validity of the results and how those threats were reduced can be found in the respective parts of this thesis.

6 Contributions

The contributions of this thesis can also be divided into two parts. Detailed conclusions of each of the papers in this thesis can be found at the end of each of the respective papers, but in this section we will summarize the main contributions of this thesis.

6.1 Sensitivity of Software Reliability

Part I of this thesis presents the derivation and evaluation of a software reliability model that can be used to evaluate the effect of changes to the usage profile of a software system on the estimated reliability of the system.

The first paper presents the theoretical derivation of the basic model and shows that this model can successfully be applied to a small fictive example. The paper also presents the reasons for the application of this analysis and derives how the results can be calculated directly without resorting to a lengthy Monte Carlo simulation. At the same time, we use the Monte Carlo simulation on the small example to validate our results and to illustrate the limitations of the developed models.

The second paper extends this basic model to include different types of changes to the usage profile and to more general types of systems. This paper makes the results of the first paper much more generally applicable to a wider range of systems and situations. The validity of these extended results is shown by again comparing the results to results from simulations on variations of the example from the first paper. The main conclusions from the first two papers is that the presented models can be used to quickly analyse the sensitivity of the reliability a software system to changes in its usage profile, without using extensive simulations. The disadvantages of this approach are that a detailed usage profile is needed, that the nature of the changes to the usage profile can be hard to predict and that when the changes are too large, the approximate models in these two papers can no longer be applied but more extensive simulations need to be used. In Table 1.1 the formulas for the different situations and for both models are summarized. The full explanation of the notation can be found in Paper II.

The third paper, on the other hand, shows the applicability of the results from the first two papers to a larger real-life system, in this case a webserver at the Department of Communication Systems at Lund University. Next to a validation of the results of the first two papers, the paper also contains a thorough discussion of the difficulties of extracting a usage profile for a webserver from standard webserver logfiles. The conclusion here is that useful results can be obtained by extracting a usage profile from a webserver logfile and applying the models from the second paper. The disadvantages of this approach are that the usage profile can not be perfectly reconstructed and that a large amount of pre-processing can be necessary. Both these disadvantages are a direct consequence of the limitations of the data in the available logfiles.

6.2 Dependability at Swedish Municipalities

Part II of this thesis presents a stable basis for starting further research that can result in practical methods to help Swedish municipalities of today to deal with evaluating the dependability of their IT systems in crisis situations. A more thorough discussion of the planned further research can be found in the next section.

The main conclusions from the work in part II so far are that Swedish municipalities are experiencing a number of important problems when trying to include risk and

Changes		Probability Model (with a terminal state)
Absolute	Paired	$2 \times n \times \sigma_\delta^2 \times \sigma_s^2 \times \|V_1\|^2$
	Spread	$\sigma_\delta^2 \times \sum_{k,l} [v_{1k} \times (s_l - s_k)]^2$
Relative	Spread	$\sigma_\delta^2 \times \sum_{k,l} \left[p_{kl} \times v_{1k} \times (s_l - s_k)\right]^2$
Specific	Spread	$\sum_{k,l} [\sigma_{kl} \times v_{1k} \times (s_l - s_k)]^2$
Changes		MTTF Model (without a terminal state)
Absolute	Paired	$2 \times n \times \sigma_\delta^2 \times \sigma_{MTTF}^2 \times \|V_1\|^2$
	Spread	$\sigma_\delta^2 \times \sum_{k,l} [v_{1k} \times (MTTF_l - MTTF_k + 1)]^2$
Relative	Spread	$\sigma_\delta^2 \times \sum_{k,l} \left[p_{kl} \times v_{1k} \times (MTTF_l - MTTF_k + 1)\right]^2$
Specific	Spread	$\sum_{k,l} [\sigma_{kl} \times v_{1k} \times (MTTF_l - MTTF_k + 1)]^2$

Table 1.1: Summary of the different formulas for the variation of the reliability of the system, for both models and for different types of changes

vulnerability analyses of their IT systems in their emergency management process. The responsibilities for this matter are divided between the emergency managers, the IT personnel and personnel of those departments responsible for the critical IT systems. Although all parties do their best to fulfil their task to the best of their abilities, a higher level of dependability could be achieved with a better cooperation.

A first problem is that the division of responsibilities leaves some critical gaps on the borders between the responsibilities of different departments. This causes some important issues to be forgotten because nobody takes responsibility for them. The IT department feels their responsibility ends with the maintenance and that the more strategical dependability planning for IT systems lies with the other departments. The other departments feel they do not have sufficient technical knowledge to investigate the dependability of the IT systems, and that it is therefore not their responsibility.

Another problem lies in the communication between the IT department and the rest of the municipality. Because of a lack of a good forum of communication, both parties feel that they can not get the information they need from the other party. The IT department can not get the users to understand the threats to the IT systems, while the rest of the municipality feels that the IT department does not understand their needs and that the IT support focuses on the wrong things.

Both these problems together make that the IT department is not involved in emergency management and as a consequence the emergency managers sometimes need

to have blind faith in their IT systems that might not be as reliable, or to prepare extensive backup measures for IT systems that are sufficiently reliable. With a better communication and cooperation on these issues, IT and emergency management resources could likely be prioritised more efficiently.

7 Future Work

Future work on this research will focus on continuing the work in both separate parts of this thesis, but also on combining some of the results of both parts.

7.1 Sensitivity of Software Reliability

A first important continuation of the work in part I would be to apply the results to a wider range of systems. The main obstacle for this application is the lack of easily available usage profiles. Although usage profiles are a widely used tool in software engineering, few actual detailed usage profiles are collected. The third paper in this thesis shows that even when extensive logfiles are available, the extraction of a usage profile of a system is far from an easy task. Many opportunities for further research on the collection of usage profiles are available in this area.

Another logical next step in the research from the first part of this thesis concerns the nature of the changing usage profile over time. More research is necessary on how future changes in the usage profile of a critical system can be predicted. This is especially important when taking into account the occurrence of rare, critical events such as crisis situations typically are. It is known that the usage of an IT system can change drastically during a crisis situation (Andersson et al. 2005), not only concerning the load, but also in the shape of the underlying usage distribution. Nevertheless, little is known on how we can predict the changes to the usage in more detail. This would require a partial shift in focus of the research from the theoretical and statistical approach to a more usage-based approach to study user behaviour during crisis situations.

7.2 Dependability at Swedish Municipalities

The main goal of the research in this part was explicitly to form a basis for further research and much further research on the results of the second part of this thesis has already been planned. In the research in part II a number of problems were identified at Swedish municipalities, and further research will focus on developing and evaluating methods that Swedish municipalities can use to deal with these problems.

Because no current methods are available that are well suited for the special situation at Swedish municipalities today, a new method needs to be developed that better fits their needs. We have chosen to develop this method based on a maturity model, since this is a common technique used in process improvement today. A maturity model is a model for process improvement that includes a number of maturity levels that can

be used to evaluate and improve the capabilities of an organisation. Typically, an organisation first evaluates itself, possibly with the help of consultants, on a number of key process areas, and is then assigned a maturity level based upon this evaluation. In the next step, a number of practices necessary for reaching a higher level of maturity can be selected and goals for the next step in process improvement can be set up. This process can be repeated until a desired level of maturity is reached. The top level of maturity is usually a level where mechanisms for continuous improvement are present across the entire organisation. The most famous maturity model is probably the Capability Maturity Model Integration from the Carnegie Mellon Software Engineering Institute (SEI, 2001).

In the next phase of this research, we will start the development of a maturity model that can help municipalities to develop, evaluate and improve the processes they use to monitor the dependability of their IT systems. The first steps in the development will consist of formulating a maturity model based on the results presented in part II of this thesis. The maturity model will include ideas from maturity models already currently in use in both IT management, such as the work presented by Luftman (2003), and in safety culture, such as the work by Fleming (2001).

One of the special challenges involved is to make this maturity model to be sufficiently simple that small municipalities can apply this with their limited resources. A first important step in the development of this maturity model will be a tool that municipalities can use for self-evaluation concerning these issues.

The development of this maturity model will be a long-term research project and will include a number of successive case studies at different municipalities where parts or finally the entire maturity model will be iteratively evaluated and improved.

Another possible continuation of this research is to study the generalizability of the results to other governmental actors and even to an international setting. Many of the problems described in the second part of this thesis are likely to be common to many other organisations than Swedish municipalities, even if their responsibilities are different and their internal organisation can be substantially different. Further exploratory research is necessary before any results can be usefully generalized to a wider scope.

7.3 Combining Parts I and II

Finally, because this thesis consists of two distinct parts, a natural opportunity for continued research would consist in combining results from both parts. The results of the second part show that at this moment most Swedish municipalities do not collect the necessary usage data to use the advanced models developed in part I.

When, in the next years, Swedish emergency management actors continue to improve their processes for evaluating the dependability of IT systems in emergency situations, this might include the measurement and prediction of detailed usage profiles that can be used as input for the models described in the first part of this thesis. At this point the models developed in the first part of this thesis can be a useful tool in the

risk and vulnerability analyses conducted at various Swedish emergency management actors. Therefore another possible continuation of the work in this thesis could focus on methods and tools to help municipalities collect the necessary usage profiles from different critical systems.

Bibliography

M. Andersson, A. Bengtsson, M. Höst, and C. Nyberg. Web Server Traffic in Crisis Conditions. *Swedish National Computing Networking Workshop*, 2005.

A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1(1):11–33, 2004.

H.A. Chan. Accelerated stress testing for both hardware and software. In *Proceedings of the Annual Symposium on Reliability and Maintainability*, pages 346–351, 2004.

N. E. Fenton and S. L. Pfleeger. *Software Metrics: A Rigorous Approach*. PWS Publishing Co., 1998.

M. Fleming. Safety culture maturity model. *Offshore Technology Report, 2000/049*, 2001.

K. Goševa-Popstojanova and S. Kamavaram. Software reliability estimation under uncertainty:generalization of the method of moments. In *Proceedings of the 8th IEEE International Symposium on High Assurance Systems Engineering*, pages 209–218, 2004.

P.-O. Hallin, J. Nilsson, and N. Olofsson. Kommunal sårbarhetsanalys, 2004. KBM:s forskningsserie, nr 3.

L. Hatton. Software failures-follies and fallacies. *IEE Review*, 43(2):49–52, 1997.

J. Johansson. Risk and vulnerability analysis of large-scale technical infrastructures. *Lic.Thesis, Department of Industrial Electrical Engineering and Automation, Lund University*, 2007.

K. Kalmelid and J. Gustavsson. Inventering av kompetensbehov m.m. inom informationssäkerhet i offentlig sektor. Technical report, Rapport, Informationssäkerhets- och analysenheten, Krisberedskapsmyndigheten, 2005.

J.D. Lawrence and J.M. Gallagher. A proposal for performing software safety hazard analysis. *Reliability Engineering and System Safety*, 55(3):267–282, 1997.

H. Le Guen, R. Marie, and T. Thelin. Reliability estimation for statistical usage testing using markov chains. In *Proceedings of the 15th International Symposium on Software Reliability Engineering*, pages 54–65, 2004.

J. N. Luftman. *Managing the Information Technology Resource: Leadership in the Information Age*. Prentice-Hall, 2003.

J.D. Musa. Operational profiles in software-reliability engineering. *IEEE Software*, 10(2):14–32, 1993.

National Research Council. The Internet Under Crisis Conditions: Learning from September 11. Technical report, 2003. URL: `http://www.nap.edu/catalog/10569.html` (Last accessed November 2007).

P. G. Neumann. Risks of untrustworthiness. In *Proceedings of the 22nd Annual Computer Security Applications Conference*, pages 321–328. IEEE Computer Society, 2006.

PTS 2005. Elektroniska kommunikationer och stormen den 8-9 januari 2005 , 2005. Post&Telestyrelsen, PTS-ER-2005:9.

A. Rae, D. Jackson, P. Ramanan, J. Flanz, and D. Leyman. Critical feature analysis of a radiotherapy machine. *Reliability Engineering and System Safety*, 89(1):48–56, 2005.

H. A. Rahman, K. Beznosov, and J. R. Martí. Identification of sources of failures and their propagation in critical infrastructures from 12 years of public failure reports. In *Proceedings of the Third International Conference on Critical Infrastructures*, page 11. The International Institute for Critical Infrastructures, 2006.

C. Robson. *Real World Research: A Resource for Social Scientists and Practitioner-researchers*. Blackwell Publishers, second edition, 2002.

SEI, 2001. Capability Maturity Model Integration, 2001. URL: `http://www.sei.cmu.edu/cmmi/` (Last accessed November 2007). Carnegie Mellon Software Engineering Institute.

SEMA, 2007. URL: `http://www.krisberedskapsmyndigheten.se/` (Last accessed November 2007). Swedish Emergency Management Agency.

F. Shull, J. Singer, and D.I.K. Sjøberg (Eds.). *Guide to Advanced Empirical Software Engineering*. Springer-Verlag London Ltd, 2007.

K. Siegrist. Reliability of systems with markov transfer of control. *IEEE Transactions on Software Engineering*, 14(7):1049–1053, 1988.

I. Sommerville. *Software Engineering*. Addison-Wesley, 2005.

B. Sundelius, E. Stern, and F. Bynander. *Krishantering på svenska: Teori och praktik*. Nerenius & Santérus, Stockholm, 1997.

D. Tang and H. Hecht. An approach to measuring and assessing dependability for critical software systems. In *Proceedings of the 8th International Symposium On Software Reliability Engineering*, pages 192–202, 1997.

M.G. Thomason and J.A. Whittaker. Rare failure-state in a markov chain model for software reliability. In *Proceedings of the 10th International Symposium on Software Reliability Engineering*, pages 12–19, 1999.

C. Trammell. Quantifying the reliability of software: statistical testing based on a usage model. In *Proceedings of the 2nd IEEE International Software Engineering Standards Symposium*, pages 208–218, 1995.

J. M. Voas and K. W. Miller. Predicting software's minimum-time-to-hazard and mean-time-to-hazard for rare input events. In *Proceedings of the International Symposium on Software Reliability Engineering*, pages 229–238, 1995.

G. H. Walton and J. H. Poore. Measuring complexity and coverage of software specifications. *Information and Software Technology*, 42(12):859–872, 2000.

A. Wesslén, P. Runeson, and B. Regnell. Assessing the sensitivity to usage profile changes in test planning. In *Proceedings of the 11th International Symposium on Software Reliability Engineering*, pages 317–326, 2000.

S. M. Yacoub and H. H. Ammar. A methodology for architecture-level reliability risk analysis. *IEEE Transactions on Software Engineering*, 28(6):529–547, 2002.

R. K. Yin. *Case Study Research: Design and Methods*. SAGE Publications Ltd, 2003.

Part I

Sensitivity of Software Reliability Models to Usage Profile Changes

Paper I

Sensitivity of System Reliability to Usage Profile Changes

Kim Weyns and Per Runeson

Presented at the 5th Conference on Software Engineering Research and Practice in Sweden (SERPS'05), October 2005, Västerås

ABSTRACT

Usage profiles and component reliability are two important factors in software system reliability estimation. To assess the sensitivity of a system's reliability to the usage profile and to the reliability of its components, a Markov based system model is used. With the help of this model, the maximum sensitivity to one change or the statistical sensitivity to many independent changes can be estimated. Advantages and limitations to this approach are discussed and finally the theory is applied to an example to show its validity.

1 Introduction

Software plays an increasingly important role in today's society. Reliable software is a prerequisite for most functions in our daily life. Power supply, banking, transportation, medical care etc. depend on reliable software. Extreme events, like bad weather conditions, terror threats or diseases, may stress the society, and hence the software systems. The current status of software reliability assessments is far from good (Runeson et al. 2003); the control over the software reliability under extreme conditions is even worse. To our knowledge, the issue of assessing the software reliability for extreme conditions, is not explored to any larger extent.

The software reliability depends not only on the defects, residing in the software systems, but also on how the software is used, i.e. the usage profile (Musa 1994a). E.g. a contributing factor to the power blackout in North America, August 2003, was a software failure in an alarm system (Poulsen SecurityFocus, 2004). The "bug was triggered by a unique combination of events and alarm conditions on the equipment it was monitoring". As software reliability assessments primarily are based on the normal usage profile, extreme events are not taken into account.

In this paper, we present an initial quantitative study on the sensitivity of the reliability estimate to changes in the usage profile. The long term goal is to assess software reliability risks with respect to extreme usage conditions.

The paper is outlined as follows: after the discussion of some related work in Section 2, Section 3 discusses the theory and in Section 4 the theory is applied and checked to an example described by Poore et al. (1993).

2 Related Work

2.1 Usage Profiles and Reliability

The usage profile, or the operational profile, is the characterization of the users' operative utilization of a software system. The usage is characterized in terms of the user-initiated events and the probability for these events (Musa 1993). The usage profile is mirrored in the utilization of the software and its components. Thus, a state-based model of the software may be developed, with the probabilities from the usage profile, determining the transition probabilities of the system model. The states in the model may represent either important states in the operation of the system or system components between which controlled is passed. This approach is proposed by Cheung (1980), later refined by Siegrist (1988a) and used by Poore et al. (1993).

In order to use the system model for reliability estimation, an explicit failure state has to be added. The direct unreliability is then expressed as the probability for a failure event in each system model state or system component. The reliability of the entire system, for a given usage profile, can then be calculated from the system model (Cheung 1980).

This model can be used to model all kind of systems were a failure and a success state

are present, and where the reliability can be defined as the probability to terminate in the success state. An example of such a system could be a server where users log in for a session to execute a number of transactions and to finally conclude with a successful termination of the session.

Systems without a terminal success state where the reliability is measured as the mean time to failure (such as most monitoring systems), are discussed by Siegrist (1988b), and a sensitivity analysis of these systems will be the topic of future work.

2.2 Sensitivity Analysis

As the software reliability depends on the usage profile, the question arises how sensitive the reliability estimate is to changes or uncertainties in the usage profile. The question has been addressed from two different perspectives; 1) based on software reliability growth models (SRGM), and 2) based on Markov models.

In the first approach, the estimated parameters of the SRGM are adjusted, based on random changes in the operational profile. Musa analyses the relative error in failure intensity and the relative error in the operational profile for a single operation (Musa 1994b). Chen et al. (1994) investigate the sensitivity of the reliability estimates to errors in the operational profile with simulation. A single error is injected in the operational profile and the effects in the reliability estimates are investigated. Crespo et al. (1996) and Pasquini et al. (1996) analyse the predicted reliability growth for different operational profiles. The real reliability growth is calculated with the Nelson reliability model (Nelson 1978, Goel 1985). Wesslén et al. (2000) model the usage with a Markov model and simulates the impact on the Nelson reliability estimate, based on multiple random changes in the usage profile.

The second approach uses a Markov chain to model the system and its usage. Thus, Poore et al. (1993) analyze the sensitivity of a system's reliability to the reliability of its components. This approach is extended by Yacoub et al. (2004). Lo et al. (2005) present an analytic approach, also based on a component model, which identifies the most sensitive parameter, component reliability and transition flow.

Goševa-Popstojanova and Kamavaram (2003; 2004) discuss two different methods for sensitivity analysis based on Markov models: the method of moments and Monte Carlo simulation. Though both methods give very good results for small systems, they require a huge amount of calculation and don't scale up well for large systems. They also don't provide a good way to get a quick estimate without lengthy calculations or simulations.

3 Sensitivity analysis

In this section we will first shortly repeat some theoretical basics by Siegrist (1988a) in Subsection 3.1, then the maximal theoretical sensitivity of the reliability estimate to one change in the operational profile is discussed in Subsection 3.2. Next the statistical sensitivity to many random changes is discussed in Subsection 3.3 and finally

some limitations of the theory are further examined in Subsection 3.4.

3.1 Definitions

In this paper we will use the Markov model as proposed by Siegrist (1988a) to model a system's usage, behaviour and reliability. The states of the model can either represent system states or system components between which control is passed, the following analysis can be used in both cases.

In the model used here, there are two main assumptions. First, the Markov property means that the future behaviour of the system is determined only by the current state of the system and not by the history of the system. Secondly, this model assumes that the system contains exactly two terminal states: a success state t and a failure state f. This means that a run of the system will always terminate in one of these two states. Next to the two terminal states, the system also contains n transient states $1, 2, \ldots, n$. State 1 represents the initial state.

The dynamics of the faultless system, without the failure state, are described by a Markov chain with state space $1, 2, \ldots, n, t$, and with transition matrix P, where p_{ij} is the probability to go from state i directly to state j.

In the imperfect system, every state has a designated reliability r_i, which means it has a probability $1 - r_i$ of failing and entering the failure state f. The dynamics of the faulty system are described by a Markov chain with state space $1, 2, \ldots, n, t, f$ and with transition matrix $\hat{P}$, given as follows:

$$\begin{aligned}
\hat{p}_{ij} &= p_{ij} \times r_i \\
&\qquad \text{for } i = 1, \ldots, n \text{ and} \\
&\qquad \text{for } j = 1, \ldots, n, t \\
\hat{p}_{if} &= 1 - r_i \\
&\qquad \text{for } i = 1, \ldots, n \\
\hat{p}_{tt} &= \hat{p}_{ff} = 1 \\
\hat{p}_{ti} &= 0 \text{ for } j = 1, \ldots, n, f \\
\hat{p}_{fi} &= 0 \text{ for } j = 1, \ldots, n, t
\end{aligned}$$

The method for computing system reliability from these transition probabilities is based on standard Markov chain theory (Kleinrock 1975).

Let $\hat{Q}$ denote the restriction of the matrix $\hat{P}$ to the transient states $1, 2, \ldots, n$, so the transition matrix $\hat{P}$ without the last two rows and without the last two columns. Then the matrix

$$V = \sum_{k=0}^{\infty} \hat{Q}^k = (I - \hat{Q})^{-1} \tag{1.1}$$

is called the potential matrix of the system. Each value v_{ij} gives the number of expected visits to state j before terminating when the system is currently in state i. Since state 1 is the starting state of the system, the system's expected number of transition periods before terminating is the sum of the elements of the first row of the matrix V.

$$\text{expected \# of periods } = sum(V_1) = \sum_{j=0}^{n} V_{1j} \tag{1.2}$$

This expected number of periods can also be seen as the expected number of events causing a state change in the system during one run of the system from start to either successful termination or failure. The events can be either user actions or internal system events and are defined together with the states in the system model.

Let

$$T = \begin{pmatrix} t_1 \\ \vdots \\ t_n \end{pmatrix} = \begin{pmatrix} \hat{p}_{1t} \\ \vdots \\ \hat{p}_{nt} \end{pmatrix} \qquad \text{and} \qquad F = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} = \begin{pmatrix} \hat{p}_{1f} \\ \vdots \\ \hat{p}_{nf} \end{pmatrix}$$

be the column vectors containing the probabilities to go directly from a given state to state t or f respectively. Then the probability s_i to finally end up in terminal state t given that the systems is currently in state i, or in other words the overall chance on success starting from state i, can be calculated as follows:

$$S = \begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix} = V \times T = (I - \hat{Q})^{-1} \times T \tag{1.3}$$

In a similar way the chance x_i to finally end up in terminal state f given that the systems is currently in state i, can be calculated as follows:

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = V \times F = (I - \hat{Q})^{-1} \times F \tag{1.4}$$

Since the system must always terminate in one of the two terminal states

$$s_i = 1 - x_i \quad \textit{for} \quad i = 1, \ldots, n \tag{1.5}$$

Because we assume that state 1 is the initial state of the system, the reliability of the whole system is simply

$$s_1 = \sum_{i=1}^{n} v_{1i} \times t_i = V_1 \times T \tag{1.6}$$

with V_1 the first row of the matrix V. Siegrist (1988a) has also shown that the sensitivity of the system's reliability to the reliability r_i of a state i is bounded by the

expected number of visits to state i in the faultless system starting from state 1, which can also be written as

$$(I - Q)^{-1}_{1i}$$

where Q denotes the restriction of the transition matrix P of the faultless system to the transient states $1, 2, \ldots, n$.

In the following sections we will build further upon the theory from Siegrist (1988a) to investigate much further how changes in the transition matrix $\hat{P}$ influence the total reliability of the system.

3.2 Maximum Sensitivity

In the previous section we saw how we can calculate the reliability of a system from its state transition matrix $\hat{P}$. For some systems, small changes in the matrix $\hat{P}$ can cause relatively big changes in the reliability of the whole system. In this section we will look at how to calculate the maximum effect of one or more small changes.

To make a distinction between the old system without the small changes and the new system with the small changes, we will use an accent to indicate the variables of the new system. So we investigate the difference between the reliability s_1 of the system with transition matrix $\hat{P}$ and the reliability s'_1 of the system with transition matrix $\hat{P}'$

Let $\delta_{ij} = \hat{p}'_{ij} - \hat{p}_{ij}$ denote the change to $\hat{p}_{ij}$ and let $\delta_{Rel} = s'_1 - s_1$ denote the resulting change in the total reliability of the system.

First of all, it's important to notice that it is impossible to change only one probability $\hat{p}_{ij}$ in the matrix since the sum of the probabilities of one row has to equal 1.

Therefore, the most simple change we can make to the system is to only change one $\hat{p}_{kt}$ and the corresponding $\hat{p}_{kf}$, or in other words, to only change the probabilities of immediate success and failure. This does not change the dynamic properties of the system, only the terminating probabilities.

$$\delta_{kt} = t'_k - t_k = f_k - f'_k = -\delta_{kf} \tag{1.7}$$

With this change, we do not change the total probability to terminate from any state, just the probability to terminate to the two terminal states. Since this change has no effect on the matrix $\hat{Q}$ and therefore also not on V as defined in equation (1.1). From the equations (1.6) and (1.7) it is immediately clear then that

$$\delta_{Rel} = v_{k1} \times \delta_{kt} \ . \tag{1.8}$$

This means that the reliability changes linearly with the size of the initial change, where the slope is determined by v_{i1}, the expected number of visits to state i. This result is very logical, since because of the change we made, for every visit to the state i we have an extra δ_{it} more chance to terminate to state t.

The situation is more complex when we make changes to the matrix $\hat{Q}$. Let's first assume that we only make a change to the transition probabilities $\hat{p}_{kl}$ and $\hat{p}_{kf}$. This

means we only change the matrix $\hat{Q}$ in one location.

$$\delta_{kl} = \hat{p}'_{kl} - \hat{p}_{kl} = f_k - f'_k = -\delta_{kf} \tag{1.9}$$

It can be easily checked that the matrix V will then change in the following way:

$$v'_{ij} - v_{ij} = \frac{\delta_{kl} \times v_{ik} \times v_{lj}}{1 - \delta_{kl} \times v_{lk}} \tag{1.10}$$

Therefore the change in the reliability of the whole system is exactly

$$\delta_{Rel} = \sum_{j=1}^{n} \frac{\delta_{kl} \times v_{1k} \times v_{lj} \times t_j}{1 - \delta_{kl} \times v_{lk}} = \delta_{kl} \times v'_{1k} \times s_l \tag{1.11}$$

In a similar way we can deduce that when we change the transition probabilities $\hat{p}_{kl}$ and $\hat{p}_{kt}$, the reliability of the entire system is affected by an amount.

$$\delta_{Rel} = -\delta_{kl} \times v'_{1k} \times (1 - s_l) \tag{1.12}$$

For small changes these results can be approximated and simplified by replacing v'_{1k} by the old value v_{1k}.

This last change discussed does no longer represent a simple change in the reliability of one state or of one transition but a real change in the usage profile of the system. An even more interesting change in the usage profile occurs when we make two changes within the matrix $\hat{Q}$, to the transition probabilities $\hat{p}_{kl}$ and $\hat{p}_{km}$,

$$\delta_{kl} = \hat{p}'_{kl} - \hat{p}_{kl} = \hat{p}_{km} - \hat{p}'_{km} = -\delta_{km} \tag{1.13}$$

then the formulas become even longer, but for small delta the resulting change in system reliability can be very well approximated by

$$\delta_{Rel} = \delta_{kl} \times v_{1k} \times (s_l - s_m) \tag{1.14}$$

Or in other words, when changing the transition probabilities from a state k to the states l and m, the resulting change in reliability can be approximated by the product of the size of the original change to the transition probabilities, multiplied by the number of expected visits to state k, multiplied by the difference between the chance of overall success starting from the states l and m. This is equal to the results found in formulas (1.11) and (1.12), when we consider that the chance of overall success starting from the states f and t is respectively 0 and 1.

This also means that the reliability is most sensitive to changes in the transition probability $\hat{p}_{kl}$, from the state k that has the higher number of expected visits, to the state l with the highest total chance of leading to success.

3.3 Statistical Sensitivity

In this section we look for the effect of a large number of small changes or uncertainties in the usage profile on the system's overall reliability. To model these changes we will assume that every transition probability $\hat{p}_{ij}$ in the system changes with a δ_{ij}. Further we will assume that all the δ_{ij} have the same distribution with mean 0 and variation Var_δ.

$$E(\delta_{ij}) = 0 \qquad \text{Var}(\delta_{ij}) = \text{Var}_\delta \tag{1.15}$$

Because we can not change one transition probability alone, for every change δ_{ij} we will select a random transition probability $\hat{p}_{ik}$ on row i that will undergo the opposite change $\delta_{ik} = -\delta_{ij}$. This will guarantee us that the sum of the transition probabilities from each state is always equal to 1.

From formula 1.14 we know that the resulting change in reliability will have a distribution with the following mean and variance:

$$E(\delta_{Rel}) = E(\delta_{ij}) \times v_{1i} \times (s_j - s_k) = 0 \tag{1.16}$$

$$\text{Var}(\delta_{Rel}) = \text{Var}(Rel) = \text{Var}_\delta \times v_{1i}^2 \times (s_j - s_k)^2 \tag{1.17}$$

Let Var_{difS} be the variation of the difference between the total chance to reach terminal state t for two randomly selected states

$$\text{Var}_{difS} = \text{Var}(s_i - s_j), \qquad i, j = 1, \ldots, n \tag{1.18}$$

Then the total change in system reliability resulting from randomly changing all the elements $\hat{p}_{ij}$ as described above, will be the sum of all the changes resulting from n changes on each of the n rows. And therefore the total change in reliability will have a distribution with the following variance:

$$\begin{aligned} \text{Var}(Rel) &= \sum_{i=1}^{n} n \times \text{Var}_\delta \times \text{Var}_{difS} \times v_{1i}^2 \\ &= n \times \text{Var}_\delta \times \text{Var}_{difS} \times \|V_1\|^2 \end{aligned} \tag{1.19}$$

Because the total change in reliability is the sum of a large number of independent changes, the distribution will be close to a normal distribution with mean 0 and a variance as defined in equation (1.19). Thus, for the standard deviation $\sigma(Rel)$ of the reliability the following holds:

$$\sum V_1 \leq \frac{\sigma(Rel)}{\sigma_\delta \times \sigma_{difS}} \leq n \times max(V_1) \tag{1.20}$$

In other words, the standard deviation of the total system reliability under small changes divided by the size of the changes and divided by the standard deviation of the difference between the total chance of success between two random states, lies

between the total number of expected periods of the system and n times the maximal number of expected visits to any state.

Equation 1.19 also shows that to decrease the sensitivity of the reliability to usage profile changes, it is important to decrease the differences between the overall success rates of the different states.

When we also want to account for random changes in the transition probabilities in the vectors T and F, equation (1.18) for Var_{difS} can simply be extended to include the states t and f (in either the index i or j, but not in both) with $s_t = 1$ and $s_f = 0$. This will of course seriously increase Var_{difS} and therefore also the variation on the reliability.

3.4 Limitations

It is important to understand that equation (1.19) only holds under a number of assumptions. First of all the changes to the system's transition probabilities have to be sufficiently small, for equation 1.14 to be a good approximation of their effect on the system's reliability. When the changes become too large the structural changes in the system model will become more important and influence the reliability of the system.

Secondly this model also assumes that positive and negative changes are equally likely for each of the transition probabilities. In practical examples this is rarely true. For example, most systems will contain a large number of transition probabilities equal to zero to indicate impossible transitions. The error on this transition probability can only be negative or zero. When we apply random changes to a system with many zeroes in its transition matrix, but without altering the zeroes, then the expected change of the reliability will no longer be zero but be biased towards a negative or positive change depending on the structure of the system. And also the estimate of the variation from equation (1.19) will be less exact. However, in most cases, it will still be a good approximation of the variation of the reliability or at least a good indication of the magnitude of the expected reliability change.

4 Example System

In this section we will apply the theory from Section 3 to an example system from Poore et al. (1993). The transition graph of the faultless system can be seen in Figure 1.1. The transition matrix and different statistics of the different states can be found in Tables 1.1 and 1.2.

The reliability of the whole system is $s(1) = 0.9899$ and the expected number of periods is $\sum v_{1i} = 67.056$.

4.1 Maximum sensitivity

From Table 1.2 we can see that the state with the most expected visits is state 5, closely followed by the states 2 and 4, the least visited state is state 7. This means the system

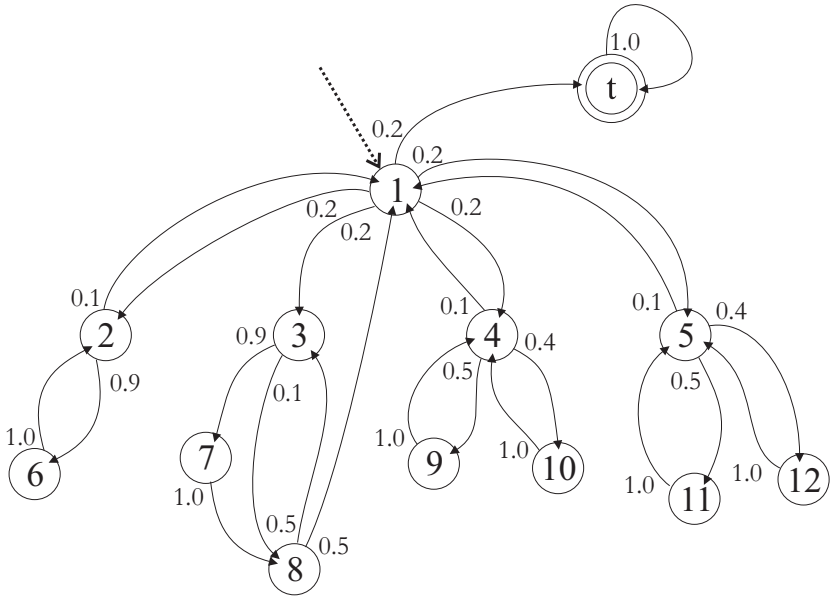

Figure 1.1: Example system. Each directed arc, labelled with its transition probability, indicates that control passes from one component to another in the direction of the arrow.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*t*
1	0	.2	.2	.2	.2	0	0	0	0	0	0	0	.2
2	.1	0	0	0	0	.9	0	0	0	0	0	0	0
3	0	0	0	0	0	0	.9	.1	0	0	0	0	0
4	.1	0	0	0	0	0	0	0	.5	.4	0	0	0
5	.1	0	0	0	0	0	0	0	0	0	.5	.4	0
6	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1	0	0	0	0	0
8	.5	0	.5	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0	0	0	0	0	0
10	0	0	0	1	0	0	0	0	0	0	0	0	0
11	0	0	0	0	1	0	0	0	0	0	0	0	0
12	0	0	0	0	1	0	0	0	0	0	0	0	0
t	0	0	0	0	0	0	0	0	0	0	0	0	1

Table 1.1: The transition matrix P of the faultless example system

i	Reliability r_i	Expected number of visits v_{1i}	Overall chance on success s_i	$(s_1 - s_i)/10^{-3}$
1	0.9990	4.9546	0.9899	0
2	0.9999	9.8895	0.9888	1.0779
3	0.9990	1.9778	0.9879	2.0154
4	0.9999	9.8899	0.9889	1.0384
5	0.9999	9.8903	0.9889	0.9890
6	1	8.8996	0.9888	1.0878
7	1	1.7783	0.9889	1.0275
8	1	1.9758	0.9889	1.0176
9	1	4.9444	0.9889	1.0483
10	1	3.9555	0.9889	1.0384
11	1	4.9447	0.9889	0.9890
12	1	3.9557	0.9889	0.9890

Table 1.2: Values of the state reliability r_i, the number of expected visits v_{1i}, the overall chance of success s_i and the difference between the overall chances of success $(s_1 - s_i)/10^{-3}$ for the different states

$\hat{p}_{kl} \overset{-\delta_{kl}=\delta_{km}}{\longrightarrow} \hat{p}_{km}$	real $\delta_{Rel}/10^{-3}$	predicted $\delta_{Rel}/10^{-3}$
$\hat{p}_{1t} \overset{0.01}{\longrightarrow} \hat{p}_{1f}$	-49.5	-49.5
$\hat{p}_{51} \overset{0.01}{\longrightarrow} \hat{p}_{5f}$	-89.1	-97.9
$\hat{p}_{13} \overset{0.01}{\longrightarrow} \hat{p}_{11}$	0.100	0.100
$\hat{p}_{78} \overset{0.01}{\longrightarrow} \hat{p}_{73}$	-0.018	-0.018
$\hat{p}_{51} \overset{0.01}{\longrightarrow} \hat{p}_{53}$	-0.199	-0.199
$\hat{p}_{51} \overset{0.05}{\longrightarrow} \hat{p}_{53}$	-0.996	-0.997
$\hat{p}_{51} \overset{0.001}{\longrightarrow} \hat{p}_{53}$	-0.020	-0.020

Table 1.3: Some real values compared with some calculated values for the change in reliability for different changes in the transition probabilities

is most sensitive to changes to the transition probabilities from these states. The state with the highest overall chance on success is state 1, the state with the lowest overall chance on success is state 3.

In Table 1.3 we can see the real effect of some transition probability changes to the overall reliability. We see that the predicted value corresponds well to the real value in all cases. For the first example the predicted value is exact, since there are no changes to the dynamic properties of the system. For the second example we see a very large change with also an error on the prediction of about 10%. This can be explained because the large change to the dynamics of the system that this change causes. A change of 0.01 is a big change for a value of $\hat{p}_{51} = 0.10$.

Overall we can see that the maximum change in reliability is about 9 times bigger than the original change to the transition probabilities. But when we do not change the values $r(i)$ and only make changes inside the matrix $\hat{Q}$, then the resulting change in reliability is only 2% of the original change to the transition probabilities because of the small differences between the overall success rates of the different states. And in that case the difference between the predicted and the real value is a lot smaller, only for very big changes the predicted value deviates a little from the real value.

4.2 Statistical Sensitivity

To check the statistical sensitivity of the system's reliability to many small changes in the transition probabilities, we will conduct a Monte Carlo simulation where we make a small random change to each of the 144 transition probabilities in the matrix $\hat{Q}$, as described in Section 3.3. To predict the effect on the system's reliability we first calculate

$$\mathrm{Var}_{difS} = \mathrm{Var}(s_i - s_j) = 5.8610^{-4}$$

and

$$\|V_1\| = 22.10\ .$$

With equation (1.19) we can now predict the standard deviation of the systems reliability for 144 random changes with $\sigma_\delta = 0.005$:

$$\sigma(Rel) = 2,24 \times 10^{-4}.$$

The resulting reliability change from 1000 random simulations, plotted on normal probability paper, can be seen in Figure 1.2. As you can see the results of the experiments fit extremely well with the predicted changes indicated by the dashed line. The measured standard deviation equals 2.32×10^{-4} while the measure average equals 1.09×10^{-6}, which is very close to the predicted 0.

4.3 Limitations

As we discussed in Section 3.4, the formulas that predict the statistical change in reliability become less precise when the σ_δ increases. For example in Figure 1.3 we

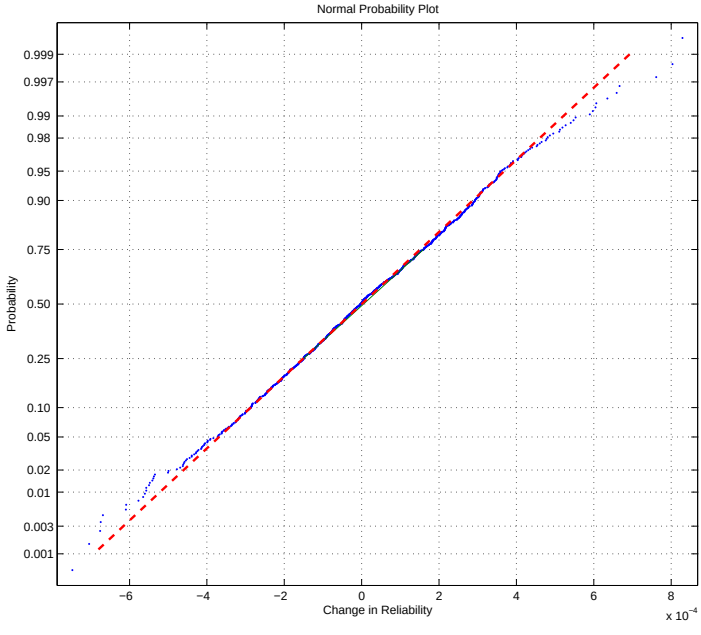

Figure 1.2: Cumulative normal plot of the predicted *(dashed line)* and experimental *(1000 dots)* statistical sensitivity to 144 random changes with $\sigma_\delta = 0.005$ from 1000 experiments

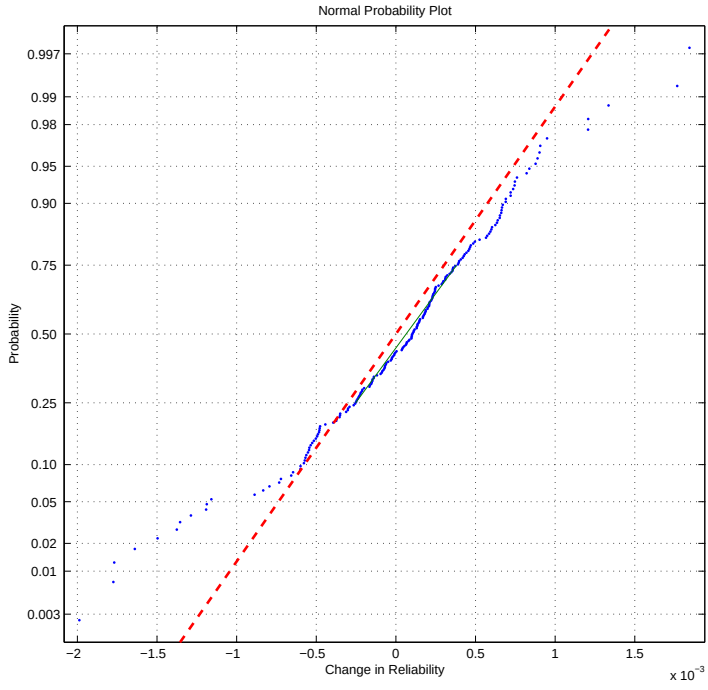

Figure 1.3: Cumulative normal plot of the predicted *(dashed line)* and experimental *(200 dots)* statistical sensitivity to 144 random changes with $\sigma_\delta = 0.01$ from 200 experiments

can see what happens when we increase σ_δ to 0.01. For 90% of the experiments, our predictions are still quite accurate, but we also notice a large group of outliers where the random changes have a large influence on the systems dynamic behaviour with a large change in the reliability as a consequence. Of course the number of outliers will increase even more as σ_δ increases.

Also, it is important to notice that in the simulation in Figure 1.1 both positive and negative changes were allowed to all transition probabilities. This means that the changed transition probability matrix $\hat{P}'$ will contain some slightly negative elements which is practically impossible. When we change the simulation and only allow positive changes to the zeros in the transition probability matrix we get the results shown in Figure 1.4. It is immediately clear that the average change in reliability is no longer 0, almost in all the experiments the change in reliability is negative. From 1000 random experiments, we find that the average change in reliability is -5.76×10^{-4}. This is due to the fact that forcing positive changes onto the zeros causes automatic negative changes to the positive values already present in the transition probability matrix.

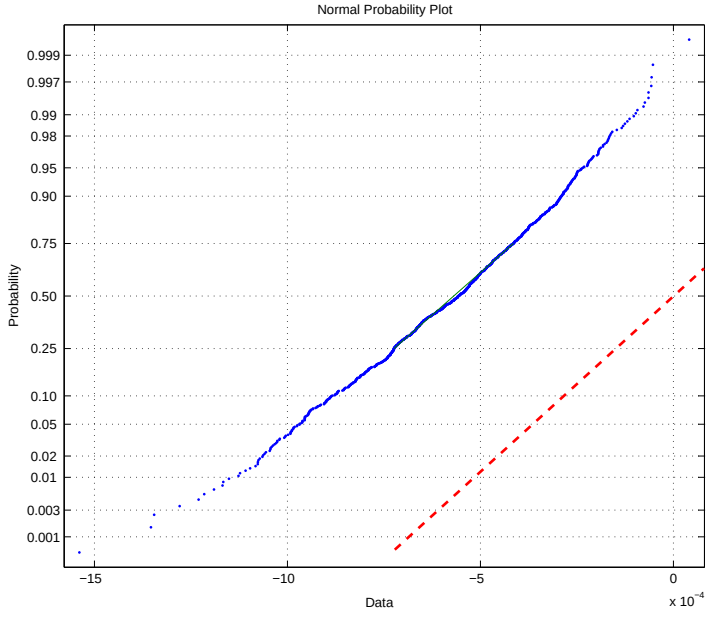

Figure 1.4: Cumulative normal plot of the predicted *(dashed line)* and experimental *(1000 dots)* statistical sensitivity to 144 random changes with $\sigma_\delta = 0.005$ from 1000 experiments, where negative changes to the zeros in $\hat{P}$ are not allowed. Although the estimate of the variation is still quite accurate, the average is clearly no longer 0, but significantly negative.

Column 1 of this matrix contains the most non-zero elements and will therefore be the most affected by these negative changes. Unfortunately state 1 is the state with the highest overall success rate. Therefore those negative changes in column 1 will have a negative effect on the reliability. From the data in Figure 1.4 we find an experimental standard deviation of 2.27×10^{-4}, which is still very close to the predicted standard deviation of 2.24×10^{-4}.

When we would simply not allow any changes to the zeros in the transition probability matrix, considering those transitions as absolutely impossible, then the variation of the system's reliability would of course be lower than the predicted value, since there are fewer changes made to the transition probability matrix which is usually quite sparse. But at least then the average change in reliability would of course still be zero, since for every change the opposite change is equally likely again.

5 Summary and Future Work

In this paper we have made a first quantitative study of the sensitivity of the reliability estimate to changes in the usage profile with the help of Markov models. With the theory described here, it is possible to make a good estimate of the effect of one or many small changes in the usage profile on the reliability of a system. Further the theory also makes it possible to easily find the transitions and states to which the reliability is most sensitive and to identify measures that can be taken to reduce this sensitivity.

When comparing the theory with some experimental results, the results are very good, taking into account a number of limitations described at the end of the paper.

Further work will be done on finding similar results for the alternative Markov model used by Siegrist (1988b), and on improving the theory to take into account the limitations posed by systems with sparse transition probability matrices. Also we will be looking more into the exact nature of the changes and uncertainties in the usage profile of different systems, with a special focus on crisis situations. A very important goal is also to apply this sensitivity analysis to a real industrial system to further investigate the applicability of these results.

Bibliography

M.-H. Chen, A. P. Mathur, and V. Rego. A case study to investigate sensitivity of reliability estimates to errors in operational profile. In *Proceedings of the 5th International Symposium on Software Reliability Engineering*, pages 276–281, 1994.

R. C. Cheung. A user-oriented software reliability model. *IEEE Transactions on Software Engineering*, 6(2):118, 1980.

A. N. Crespo, P. Matrella, and A. Pasquini. Sensitivity of reliability growth models to operational profile errors. In *Proceedings of the 7th International Symposium on Software Reliability Engineering*, pages 35–44, 1996.

A. L. Goel. Software Reliability Models: Assumptions, Limitations, and Applicability. *IEEE Transactions on Software Engineering*, SE11(12):1411–1424, 1985.

K. Goševa-Popstojanova and S. Kamavaram. Assessing uncertainty in reliability of component-based software systems. In *Proceedings of the 14th International Symposium on Software Reliability Engineering*, pages 307–320, 2003.

K. Goševa-Popstojanova and S. Kamavaram. Software reliability estimation under uncertainty:generalization of the method of moments. In *Proceedings of the 8th IEEE International Symposium on High Assurance Systems Engineering*, pages 209–218, 2004.

L. Kleinrock. *Queueing systems*, volume 1: Theory. Wiley, New York, 1975.

J.-H. Lo, C.-Y. Huang, I.-Y. Chen, S.-Y. Kuo, and M.R. Lyu. Reliability assessment and sensitivity analysis of software reliability growth modeling based on software module structure. *Journal of Systems and Software*, 76(1):3–13, 2005.

J.D. Musa. Adjusting measured field failure intensity for operational profile variation. In *Proceedings of the 5th International Symposium on Software Reliability Engineering*, pages 330–333, 1994a.

J.D. Musa. Operational profiles in software-reliability engineering. *IEEE Software*, 10(2):14–32, 1993.

J.D. Musa. Sensitivity of field failure intensity to operational profile errors. In *Proceedings of the 5th International Symposium on Software Reliability Engineering*, pages 334–337, 1994b.

E. Nelson. Estimating software reliability from test data. *Microelectronics and Reliability*, 17:67–74, 1978.

A. Pasquini, A.N. Crespo, and P. Matrella. Sensitivity of reliability-growth models to operational profile errors vs. testing accuracy. *IEEE Transactions on Reliability*, 45 (4):531–540, 1996.

J. H. Poore, H. D. Mills, and D. Mutchler. Planning and certifying software system reliability. *IEEE Software*, 10(1):88–99, 1993.

K. Poulsen. Software bug contributed to blackout. SecurityFocus, 2004. URL: `http://www.securityfocus.com/news/8016` (Last accessed November 2007).

P. Runeson, C. Andersson, and M. Höst. Test processes in software product evolution-a qualitative survey on the state of practice. *Journal of Software Maintenance and Evolution: Research and Practice*, 15(1):41–59, 2003.

K. Siegrist. Reliability of systems with markov transfer of control. *IEEE Transactions on Software Engineering*, 14(7):1049–1053, 1988a.

K. Siegrist. Reliability of systems with markov transfer of control, ii. *IEEE Transactions on Software Engineering*, 14(10):1478–1480, 1988b.

A. Wesslén, P. Runeson, and B. Regnell. Assessing the sensitivity to usage profile changes in test planning. In *Proceedings of the 11th International Symposium on Software Reliability Engineering*, pages 317–326, 2000.

S. Yacoub, B. Cukic, and H. H. Ammar. A scenario-based reliability analysis approach for component-based software. *IEEE Transactions on Reliability*, 53(4):465–480, 2004.

Paper II

Sensitivity of Software System Reliability to Usage Profile Changes

Kim Weyns and Per Runeson

Published in the proceedings of the 22nd Annual ACM Symposium on Applied Computing, March 2007, Seoul, Korea

ABSTRACT

Usage profiles are an important factor in software system reliability estimation. To assess the sensitivity of a system's reliability to changes in the usage profile, a Markov based system model is used. With the help of this model, the statistical sensitivity to many independent changes can be estimated. The theory supports both absolute and relative changes and can be used for systems with or without a terminal state. With this approach it is possible to very quickly estimate the uncertainty on the predicted reliability calculated from a Markov model based upon the uncertainty on the usage profile. Finally the theory is applied to an example to illustrate its use and to show its validity.

1 Introduction

Software plays an increasingly important role in today's society. Reliable software is a prerequisite for most functions in our daily life: power supply, transportation, medical care etc... The software reliability depends not only on the defects residing in the software system, but also on how the software is used, i.e. the usage profile (Musa 1994a). It is often very hard to determine this usage profile exactly and usually it changes over time. However, when the reliability of a software system is estimated, an uncertainty on this usage profile is rarely taken into account.

In this paper we present a quantitative study on the sensitivity of the reliability estimate to changes in the usage profile. After the discussion of some related work in Section 2, Section 3 discusses the theory and in Section 4 this theory is applied to an example from Poore et al. (1993). In Section 5 we adapt our theory for a different class of systems. Finally, in Section 7 the conclusions of this paper are summarized.

2 Related Work

2.1 Usage Profiles and Reliability

The usage profile, or the operational profile, describes the users' utilization of a software system in terms of the user-initiated events and the probability for these events (Musa 1993). The usage profile is mirrored in the utilization of the software and its components. Thus, a state-based Markov model of the software may be developed, with the probabilities from the usage profile, determining the transition probabilities of the system model. This approach is proposed by Cheung (1980) and later refined by Siegrist (1988a).

In order to use the system model for reliability estimation, an explicit failure state has to be added. The direct unreliability is then expressed as the probability for a failure in each system model state. The reliability of the entire system, for a given usage profile, can then be calculated from the system model (Cheung 1980).

This model can be used to model systems where the reliability can be defined as the probability to terminate in the success state and not the failure state. An example of such a system could be a server where users log in for a session to execute a number of transactions and to finally conclude with a successful termination of the session.

For systems without a terminal success state, the reliability is usually expressed as the mean time to failure. In this paper we first limit the discussion to systems with a terminal success state and then describe how the results can be adapted for systems without such a state in Section 5.

2.2 Sensitivity Analysis

As the software reliability depends on the usage profile, the question arises how sensitive the reliability estimate is to changes or uncertainties in the usage profile. The

question has been addressed from two different perspectives: based on software reliability growth models (SRGM) (Musa 1994b, Pasquini et al. 1996, Wesslén et al. 2000), and based on Markov models.

With Markov models Poore et al. (1993) and Yacoub et al. (2004) analyze the sensitivity of a system's reliability to the reliability of its components. Lo et al. (2005) present an analytic approach which identifies the most sensitive parameter, component reliability and transition flow.

The sensitivity to many changes in the transition probabilities has been discussed by Goševa-Popstojanova and Kamavaram (2004) with two methods based on Markov models: the method of moments and Monte Carlo simulation. Both methods give very good results for small systems, but require a large amount of calculation and hence don't scale up well for large systems.

Our method is a simplification of the method of moments, and in Subsection 3.6 we discuss shortly how our results can be derived from the more general formulas by Goševa-Popstojanova and Kamavaram (2004). The advantages of our method are that our method is simpler, requires less computation and does not require the explicit estimation of a huge number of covariances between different transition probabilities. This makes our method more practical to use, especially for large systems. The method of moments has the advantage that it is more general and gives more exact results when all the covariances are known. In Section 4 we use the Monte Carlo simulation described by Goševa-Popstojanova and Kamavaram (2004) to check the validity of our results on a small example.

3 Sensitivity analysis

In this section we first shortly repeat some theoretical basics from Siegrist (1988a), then the maximal theoretical sensitivity of the reliability estimate to one change in the operational profile is discussed in Subsection 3.2. Next the statistical sensitivity to many random changes is discussed and some limitations and solutions are further examined in Subsections 3.4 and 3.5. Finally Subsection 3.6 discusses the relationship to the results by Goševa-Popstojanova and Kamavaram (2004).

3.1 Definitions

We use the Markov model as proposed by Siegrist (1988a) to model a system's usage and reliability. The states of the model can either represent system states or system components between which control is passed.

In the model used here, there are two main assumptions. First, the Markov property means that the future behaviour of the system is determined only by the current state of the system and not by the history of the system. Secondly, this model assumes that the system contains exactly two terminal states: a success state t and a failure state f. This means that a run of the system always terminates in one of these two states. In addition to the two terminal states, the system also contains n transient

states $1, 2, \ldots, n$. State 1 represents the initial state.

The dynamics of the faultless system, without the failure state, are described by a Markov chain with state space $1, 2, \ldots, n, t$, and with transition matrix P, where p_{ij} is the probability to go from state i directly to state j.

In the imperfect system, every state has a designated reliability r_i, which means it has a probability $1 - r_i$ of directly entering the failure state f. The dynamics of the faulty system are described by a Markov chain with state space $1, 2, \ldots, n, t, f$ and with transition matrix $\hat{P}$, given as follows:

$$\begin{array}{ll} \hat{p}_{ij} = p_{ij} \times r_i & \hat{p}_{if} = 1 - r_i \\ \hat{p}_{tt} = \hat{p}_{ff} = 1 & \hat{p}_{tf} = \hat{p}_{ft} = \hat{p}_{ti} = \hat{p}_{fi} = 0 \end{array}$$

$$\text{for } i = 1, \ldots, n \text{ and for } j = 1, \ldots, n, t$$

The method for computing system reliability from these transition probabilities is based on standard Markov chain theory (Kleinrock 1975). Let $\hat{Q}$ denote the restriction of the transition matrix $\hat{P}$ to the transient states $1, 2, \ldots, n$, so the transition matrix $\hat{P}$ without the last two rows and without the last two columns. Then the matrix $V = (I - \hat{Q})^{-1}$ is called the potential matrix of the system. Each value v_{ij} gives the number of expected visits to state j before terminating when the system is currently in state i. Since state 1 is the starting state of the system, the system's expected number of transition periods before terminating is the sum of the elements of the first row of the matrix V.

Let $T = (t_i)$ and $F = (f_i)$ be the column vectors containing the probabilities to go directly from a given state to state t or f respectively. Then the probability s_i to finally end up in terminal state t given that the system is currently in state i, or in other words the overall chance on success starting from state i, can be calculated as follows:

$$S = \begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix} = V \times T = (I - \hat{Q})^{-1} \times T \tag{2.1}$$

Because we assume that state 1 is the initial state of the system, the reliability of the whole system, is simply s_1. For each of the states the following equality holds:

$$s_i = \sum_{j=1}^{n} s_j \times \hat{p}_{ij} + t_i \tag{2.2}$$

Siegrist (1988a) uses this formula is to calculate a maximum to the sensitivity of the system's reliability to the direct reliability r_i of a state i. In the following sections we build further upon this theory to investigate much further how changes in $\hat{P}$ influence the total reliability of the system.

3.2 Sensitivity to One Change

To make a distinction between the old system without the small changes and the new system with the small changes, we use an accent to indicate the variables of the new system. So we investigate the difference between the old reliability s_1 of the system with transition matrix $\hat{P}$ and the changed reliability s_1' of the system with changed transition matrix $\hat{P}'$.

Let $\delta_{kl} = \hat{p}'_{kl} - \hat{p}_{kl}$ denote the change to $\hat{p}_{ij}$ and let $\delta_{Rel} = s_1' - s_1$ denote the resulting change in the total reliability of the system.

First of all, it's important to notice that it is impossible to change only one probability $\hat{p}_{kl}$ in the matrix since the sum of the probabilities of one row has to equal 1. One possible solution is to always make two opposite changes within the matrix $\hat{Q}$, to two transition probabilities $\hat{p}_{kl}$ and $\hat{p}_{km}$,

$$\delta_{kl} = \hat{p}'_{kl} - \hat{p}_{kl} = \hat{p}_{km} - \hat{p}'_{km} = -\delta_{km}. \tag{2.3}$$

For small changes the resulting change in system reliability can then be approximated by

$$\delta_{Rel} \approx \delta_{kl} \times v_{1k} \times (s_l - s_m) \tag{2.4}$$

This formula also holds when the states l or m are one of the terminal states, when we consider that the chance of success starting from the states f and t is respectively 0 and 1.

This also means that the reliability is most sensitive to changes in the transition probability $\hat{p}_{kl}$, from the state k that has the higher number of expected visits, to the states with the highest and lowest total chance of leading to success. This corresponds to the findings published by Lo et al. (2005).

Instead of always making two opposite changes, another option is to make a change δ_{kl} to the transition probability $\hat{p}_{kl}$ and then to rescale the whole row so the sum becomes 1 again. This method we call *spread changes* as opposed to the *paired changes* described above. This is equivalent to making $n+1$ paired changes of size $\frac{\delta_{kl} \cdot \hat{p}_{ki}}{1+\delta_{kl}}$ from the probabilities $\hat{p}_{ki}$ ($i \in \{1, ..., n, t, f\} \setminus \{l\}$) to $\hat{p}_{kl}$.

When adding together the effect of all these changes and ignoring second order effects, we can approximate the change in the reliability of the whole system caused by such a spread change as

$$\delta_{Rel} \approx \frac{\delta_{kl}}{1 + \delta_{kl}} \times v_{1k} \times (s_l - s_k) \tag{2.5}$$

This formula follows directly from equations (2.4) and (2.2). For small changes we can ignore the denominator $1 + \delta_{kl}$ and still retain a good approximation. This will simplify the statistical sensitivity analysis in the following sections.

3.3 Statistical Sensitivity to Absolute Changes

In this section we look for the effect of a large number of small changes or uncertainties in the usage profile on the system's overall reliability. First we look at the

most simple case. Here we assume that all the δ_{kl} have an identical and independent distribution with mean 0 and variation σ^2_δ. This is equivalent to assuming an equal absolute uncertainty on all the transition probabilities.

$$E(\delta_{kl}) = 0, \qquad \sigma^2(\delta_{kl}) = \sigma^2_\delta, \qquad k,l = 1,\ldots,n \tag{2.6}$$

Because we can not change one transition probability alone, for every change δ_{kl} we select a random transition probability $\hat{p}_{km}$ on row k that will undergo the opposite change $\delta_{km} = -\delta_{kl}$. This will guarantee that the sum of the transition probabilities from each state is always equal to 1.

From standard statistics we know that:

$$\sigma^2(s_i - s_j) = 2 \times \sigma^2(s) = 2 \times \sigma^2_s, \qquad i,j = 1,\ldots,n \tag{2.7}$$

Now the total change in system reliability resulting from randomly and independently changing all the elements $\hat{p}_{kl}$ as described above, is the sum of all the changes resulting from n independent changes on each of the n rows. And therefore the total change in reliability will have a distribution with the following variance:

$$\sigma^2(Rel) \approx 2 \times n \times \sigma^2_\delta \times \sigma^2_s \times \|V_1\|^2 \tag{2.8}$$

Because the total change in reliability is the sum of a large number independent small changes, the distribution is close to a normal distribution with mean 0.

When we also want to account for random changes in the transition probabilities in the vectors T and F, equation (2.7) for σ^2_s can simply be extended to include the states t and f with $s_t = 1$ and $s_f = 0$. This would of course seriously increase σ^2_s and therefore also the variation on the reliability.

Another option is to apply a random spread change to each of the transition probabilities of the matrix and all the δ_{ij} have the same distribution with mean 0 and variation σ^2_δ. Because the different factors in equation (2.5) are not independent we can not make the same simplification as with paired changes. In this case the standard deviation of the reliability can be approximated by

$$\sigma(Rel) \approx \sigma_\delta \times \sqrt{\sum_{k,l} [v_{1k} \times (s_l - s_k)]^2} \quad k,l = 1,\ldots,n \tag{2.9}$$

3.4 Limitations

It is important to understand that the equations (2.8) and (2.9) only hold under a number of assumptions. First of all the changes to the system's transition probabilities have to be sufficiently small, for equation (2.4) and (2.5) to be a good approximation. When the changes become too large, the structural changes in the system model become more important and influence the reliability of the system.

Secondly this model also assumes that positive and negative changes are equally likely for each of the transition probabilities. In practical examples this is rarely true. For

example, most systems contain a large number of transition probabilities equal to zero to indicate impossible transitions. The error on this transition probability can only be negative or zero. When we apply random changes to a system with many zeroes in its transition matrix while only allowing positive changes to the zeros, then the expected change of the reliability will no longer be zero but be biased towards a negative or positive change depending on the structure of the system. But in most cases equation 2.8 will still be a good indication of the magnitude of the expected reliability change.

3.5 Statistical Sensitivity to Relative Changes

To overcome these problems we can use relative changes, which means we assume a larger absolute uncertainty on larger probabilities. This will of course also make sure that impossible transitions remain impossible. For most systems this is a much better model of the uncertainty on the transition probabilities.

Here we have to use the spread changes discussed before, since it makes no sense to make paired changes relative to the size of one of the changed probabilities without taking the size of the other probability into account.

If we assume an equal relative change or uncertainty σ_δ for all of the transition probabilities then the total standard deviation of the overall reliability can be approximated by

$$\sigma(Rel) \approx \sigma_\delta \times \sqrt{\sum_{k,l} \left[p_{kl} \times v_{1k} \times (s_l - s_k)\right]^2}$$
$$k = 1, \ldots, n \qquad l = 1, \ldots, n, (t, f). \tag{2.10}$$

When we want to introduce even more detail into the model and want to model *specific* uncertainties for different transition probabilities we can adapt the equation above in the following way:

$$\sigma(Rel) \approx \times \sqrt{\sum_{k,l} \left[\sigma_{kl} \times v_{1k} \times (s_l - s_k)\right]^2}$$
$$k = 1, \ldots, n \qquad l = 1, \ldots, n, t, f. \tag{2.11}$$

where σ_{kl} is the standard deviation (or in other words the uncertainty) on the transition probability p_{kl}.

3.6 Relation to the Method of Moments

Our formulas can be seen as a special case of the method of moments presented by Goševa-Popstojanova and Kamavaram (2004). The first order method of moments requires the calculation of all derivates of the reliability to all the transition probabilities, and further requires the estimation of about $O(n^3/2)$ covariances between the transition probabilities.

This huge number can make the method unpractical to use for systems with a large amount of states. The covariances can not simply be neglected because the sum of the

transition probabilities from each state has to equal 1 which typically creates many negative covariances between these transition probabilities.

To show the relationship between the formulas by Goševa-Popstojanova and Kamavaram (2004) and our formulas, we just have to calculate those first order derivates of the reliability and calculate the covariances implied by paired or spread changes.

In the derivation of our formulas we assumed that all $E(\delta_{kl})$ are equal to zero. This is a quite logical assumption, since we calculate the reliability with the transition probabilities that we consider most likely. With this assumption the formulas for the first order derivates of the reliability to the transition probabilities can be reduced to

$$\frac{\partial R}{\partial p_{ij}} = v_{1k} \times s_l. \tag{2.12}$$

By using paired or spread changes we are implicitly making assumptions about the covariances between the different transition probabilities. This at the same time makes the formulas less general but also more practical to use when no such detailed information is available. With standard statistical methods the new absolute variances and covariances implied by spread or paired changes can be calculated. For example, spread changes implies

$$Cov(p_{ki}, p_{kj}) = \sum_{l=1}^{n} \sigma_{kl}^2 \times p_{ki} \times p_{kj} - p_{ki} \times \sigma_{kj}^2 - p_{kj} \times \sigma_{ki}^2. \tag{2.13}$$

By incorporating these results into the results published by Goševa-Popstojanova and Kamavaram (2004), the formulas can be simplified to the results presented in formulas (2.9), (2.10) or (2.11).

4 Example System

In this section we apply the theory from Section 3 to an example system from Poore et al. (1993). The transition graph of the faultless system can be seen in Figure 2.1. The reliability is $s_1 = 0.9899$ and the expected number of periods is $\sum v_{1i} = 67.056$.

4.1 Statistical Sensitivity to Absolute Changes

With equation (2.8) we can predict the standard deviation of the systems reliability for 144 random changes with $\sigma_\delta = 0.005$ to be $\sigma(Rel) = 2.24 \times 10^{-4}$.

To check this result we conducted a Monte Carlo simulation. The resulting reliability change from 500 simulation runs can be seen in Figure 2.2. The results of the experiments fit very well with the predicted changes indicated by the dashed line. This indicates that the real standard deviation is close to the predicted value.

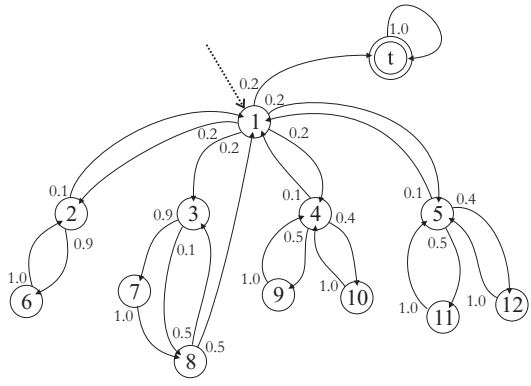

Figure 2.1: Example system without the failure state. Each directed arc, labelled with its transition probability, indicates that control passes from one component to another.

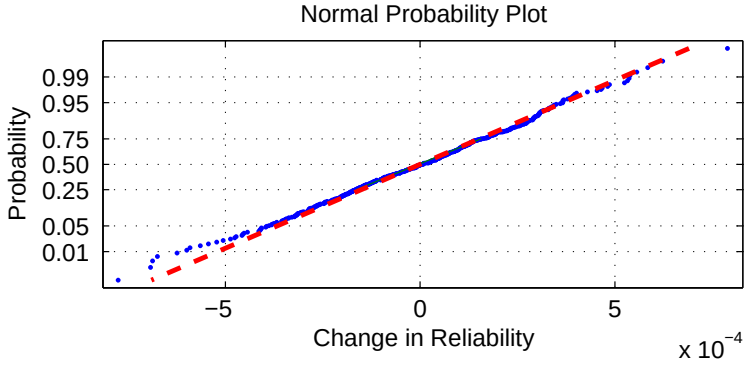

Figure 2.2: Cumulative normal plot of the predicted *(dashed line)* and experimental *(500 dots)* sensitivity to 144 random paired changes with $\sigma_\delta = 0.005$

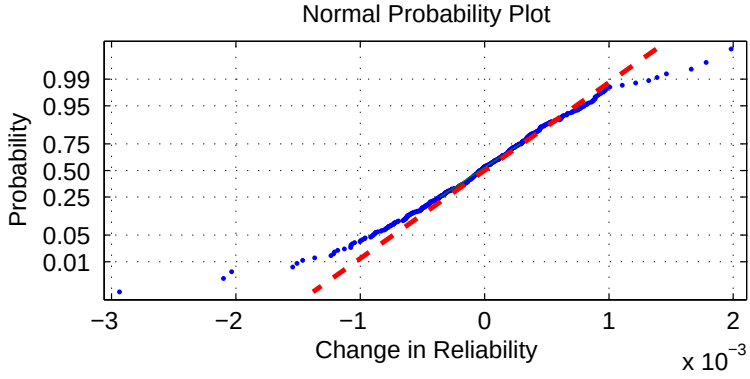

Figure 2.3: Cumulative normal plot of the predicted *(dashed line)* and experimental *(500 dots)* sensitivity to 144 random paired changes with $\sigma_\delta = 0.01$.

4.2 Limitations

As we discussed in Section 3.4, the formulas that predict the statistical change in reliability become less precise when the σ_δ increases. For example in Figure 2.3 we can see what happens when $\sigma_\delta = 0.01$. For 90% of the experiments, our predictions are still quite accurate, but we also notice a group of outliers where the changes have a large influence on the system's dynamic behaviour with a large change in the reliability as a consequence. This is due to the fact that the matrix P contains some small transition probabilities where a change of a few times 0.01 can already have a large effect.

4.3 Statistical Sensitivity to Relative Changes

The results of a similar simulation with relative changes of 10% can be seen in figure 2.4. Here the predicted uncertainty on the reliability is 3.9×10^{-4}. Working with relative changes allows us to accurately predict the sensitivity to quite large changes. When we would include an equal relative uncertainty of 10% on the transition probabilities to go to the terminal states, the predicted uncertainty on the reliability would be 1.2×10^{-3}, which is about 12% of the probability for a run of the system to terminate in the failure state.

5 Alternative System Model

In some kind of software systems there is no clear success state and ideally the system would never terminate, since the only way for the system to terminate is in a failure. For these kinds of systems, reliability is usually expressed as the time between failures of the system (Mean Time To Failure).

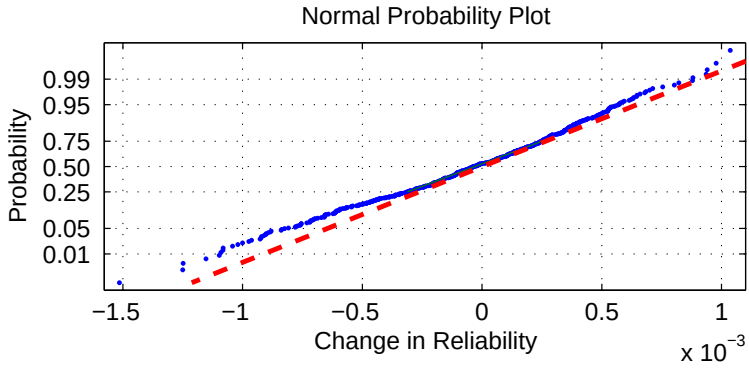

Figure 2.4: Cumulative normal plot of the predicted *(dashed line)* and experimental *(500 dots)* statistical sensitivity to 144 random spread relative changes with $\sigma_\delta = 10\%$

These systems can be modelled by a similar Markov chain, but with only one terminal state f (Siegrist 1988b). Here the mean time to failure, expressed as a number of transitions before termination in the terminal state f, given that the system is presently in state i, can be calculated as

$$MTTF_i = \sum_{j=1}^{n} v_{ij}. \tag{2.14}$$

For this model we can make a similar analysis. For example for spread, relative changes we find:

$$\sigma(Rel) \approx$$
$$\sigma_\delta \times \sqrt{\sum_{k,l} \left[p_{kl} \times v_{1k} \times (MTTF_l - MTTF_k + 1)\right]^2}$$
$$k = 1, \ldots, n \qquad l = 1, \ldots, n, (f) \tag{2.15}$$

6 Summary and Future Work

In this paper we have made a quantitative study of the sensitivity of the reliability estimate to changes in the usage profile. With this theory it is possible to make a good estimate of the effect of many small changes in the usage profile on the reliability of a system. The main advantage of this methods over the related method of moments from Goševa-Popstojanova and Kamavaram (2004) is that it does not require as much information and is computationally much simpler. The disadvantage is that it is less general and can not be used when very specific information about the covariances between the transition probabilities needs to be taken into account.

When comparing the theory with some experimental results, the results are very good, taking into account a number of limitations described in Subsection 3.4. For absolute changes the formulas only give a good approximation for small changes, but for relative changes the formulas can quickly give a realistic estimate of the uncertainty on a reliability estimate from a Markov usage model.

Further research will be done on the exact nature of the uncertainties in the usage profile of different systems. Another very important goal is also to apply this sensitivity analysis to a real operational system to further investigate the applicability of these results.

Bibliography

R. C. Cheung. A user-oriented software reliability model. *IEEE Transactions on Software Engineering*, 6(2):118, 1980.

K. Goševa-Popstojanova and S. Kamavaram. Software reliability estimation under uncertainty:generalization of the method of moments. In *Proceedings of the 8th IEEE International Symposium on High Assurance Systems Engineering*, pages 209–218, 2004.

L. Kleinrock. *Queueing systems*, volume 1: Theory. Wiley, New York, 1975.

J.-H. Lo, C.-Y. Huang, I.-Y. Chen, S.-Y. Kuo, and M.R. Lyu. Reliability assessment and sensitivity analysis of software reliability growth modeling based on software module structure. *Journal of Systems and Software*, 76(1):3–13, 2005.

J.D. Musa. Adjusting measured field failure intensity for operational profile variation. In *Proceedings of the 5th International Symposium on Software Reliability Engineering*, pages 330–333, 1994a.

J.D. Musa. Operational profiles in software-reliability engineering. *IEEE Software*, 10(2):14–32, 1993.

J.D. Musa. Sensitivity of field failure intensity to operational profile errors. In *Proceedings of the 5th International Symposium on Software Reliability Engineering*, pages 334–337, 1994b.

A. Pasquini, A.N. Crespo, and P. Matrella. Sensitivity of reliability-growth models to operational profile errors vs. testing accuracy. *IEEE Transactions on Reliability*, 45 (4):531–540, 1996.

J. H. Poore, H. D. Mills, and D. Mutchler. Planning and certifying software system reliability. *IEEE Software*, 10(1):88–99, 1993.

K. Siegrist. Reliability of systems with markov transfer of control. *IEEE Transactions on Software Engineering*, 14(7):1049–1053, 1988a.

K. Siegrist. Reliability of systems with markov transfer of control, ii. *IEEE Transactions on Software Engineering*, 14(10):1478–1480, 1988b.

A. Wesslén, P. Runeson, and B. Regnell. Assessing the sensitivity to usage profile changes in test planning. In *Proceedings of the 11th International Symposium on Software Reliability Engineering*, pages 317–326, 2000.

S. Yacoub, B. Cukic, and H. H. Ammar. A scenario-based reliability analysis approach for component-based software. *IEEE Transactions on Reliability*, 53(4):465–480, 2004.

Paper III

Sensitivity of Website Reliability to Usage Profile Changes

Kim Weyns and Martin Höst

Published in the proceedings of the 18th IEEE International Symposium on Software Reliability Engineering (ISSRE 2007), November 2007, Trollhättan, Sweden

ABSTRACT

To measure the reliability of a website from a user's point of view, the uncertainty on the usage of the website has to be taken into account. In this paper we investigate the influence of this uncertainty on the reliability estimate for a web server. For this purpose a session based Markov model is used to model the usage extracted from the server's logfiles. From these logfiles a complete user profile can be extracted together with an estimate of the uncertainty on this user profile. This paper investigates the applicability of this kind of Markov model on web server reliability and discusses the difficulties with data extraction from the logfiles. Advantages and disadvantages of this approach are discussed and the approach is applied to data from a university department's web server to demonstrate its applicability.

1 Introduction

The Internet plays an increasingly important role in today's society. We have become more and more dependent on the Internet for critical functions in our society. In the event of a crisis in our society the Internet can play an important role in the information spreading (National Research Council 2003). The most important quality attribute when accessing critical information or services on a web server is the reliability. This reliability is very hard to measure because it is a combination of reliability on the client side, on the network and on the server side (Ma and Tian 2007). In this paper we focus on faults caused by broken links between different parts of the website. For the reliability from a user perspective it is not only important how many broken links are present, but also how often they are used.

Just as for most software systems, reliability of a web server should be measured based on the usage profile. It can be hard to determine this usage profile exactly and it often changes over time, especially when the webserver is updated. In this paper we wish to take the uncertainty on this usage profile into account. Therefore we investigate how a Markov model for the usage can be extracted from the logfiles of a web server, and how it can be used to measure and predict the reliability of the web server.

2 Related Work

The reliability of a website could be defined and measured in many ways. Ma and Tian (2007) divide the failures that can occur when accessing information from a website into three categories according to their source: '*host, network or browser failures*', which are basically failures in one of the technical subsystems from content provider to the user, '*user failures*', which are faults directly caused by the user and '*source or content failures*', such as for example faulty or broken links between webpages. This last category, on which we focus in this paper, is specific for the web server domain and in this paper we try to model and measure reliability related to exactly these failures.

Sampath et al. (2004) have successfully used the logfiles of a webserver to generate usage profiles for automated testing, and Goševa-Popstojanova et al. (2006) study the logfiles to measure the reliability from a user perspective. In the log files, as in Figure 3.1 the HTTP return code of every request indicates whether the server was able to provide the data requested by the user. For example the normal return code is '200' when the content was correctly returned. When the requested document was not found, the return code is '404' and an error page was displayed. The complete list of possible return codes for HTTP version 1.1 is defined in the specification RFC 2616 (Mogul et al. 1999).

With these recorded return codes it can be measured how many of the requests were successfully answered, and Goševa-Popstojanova et al. (2006) called this the request-based reliability. Because each user session usually contains many requests, Goševa-Popstojanova et al. (2006) also define the session-based reliability, where all sessions that contain at least one failed request are considered as failed.

```
IP_userX - - [03/Aug/2006:10:23:45 +0200] "GET
    /" 200 11252 "-"
IP_userX - - [03/Aug/2006:10:23:54 +0200] "GET
    /education/" 200 20608 "http://serg.telecom.lth.se/"
IP_userX - - [03/Aug/2006:10:24:18 +0200] "GET
    /education/metodik/" 200 13190
    "http://serg.telecom.lth.se/education/"
IP_userX - - [03/Aug/2006:10:24:23 +0200] "GET
    /education/metodik/old_exam.html" 200 9666
    "http://serg.telecom.lth.se/education/metodik/"
IP_userX - - [03/Aug/2006:10:24:38 +0200] "GET
    /education/metodik/T2005.pdf" 200 17295
    "http://.../education/metodik/old_exam.html"
```

Figure 3.1: Example session from the logfile

The main research question in this paper is how sensitive the session-based reliability of a website is to the uncertainty on the usage. Another question we try to answer is how a change in the usage, for example because of the changing structure or content of the website, influences the reliability.

To study this sensitivity we use the techniques described by the authors of this paper in (Weyns and Runeson 2007) for sensitivity analysis of software reliability modelled by Markov chains, which at the same time gives us the opportunity to test the applicability of these methods on a much larger system than before.

3 Data Processing

3.1 Important Issues

Most webservers automatically log all requests to logfiles. Before this logged data can be used to build a usage profile for the web server that can be used in reliability prediction, many issues have to be sorted out. To extract the user profile we take the following steps:

1. collect the logfiles, which are usually very big,
2. filter out web robots and irrelevant document types,
3. generate a list of documents,
4. compile the requests into user sessions,
5. generate a matrix with the amount of transitions,
6. convert the matrix to transition probabilities.

Some of the steps that deserve special attention are discussed in the following paragraphs.

Filtering

When analysing the log data from a web server it can be seen quite fast that a substantial part of the requests are not from actual users, but from all kinds of web robots, commonly known as Internet bots or simply bots, crawling the website. Since we are not interested here in the reliability that the bots experience and because they do not behave as normal users, we need to filter out these requests. This is not a trivial task and can be done in many ways. Tan and Kumar (2002) discuss both simple and more advanced techniques for robot detection. One of the most common ways is by excluding all requests from a specified list of known search bot IP's. Other types of bots can usually be found by inspecting some statistics from the logfiles to search for irregularities, such as an unusually high number of requests from the same IP address. Some bots might escape detection, but those are then only responsible for a small part of the traffic and can therefore probably be safely ignored.

Other IP addresses we might exclude from the analysis are those addresses that do not represent the user group we are interested in, such as for example the organisation responsible for the updating of the web server. If we are for example only interested in external users it is possible to exclude those IP addresses coming from local computers. This method could also be used to group the users based upon the IP address, for example by country, in case we wish to generate a number of separate user profiles.

Secondly, we need only consider the requests that concern actual pages, and not the images or other elements that are requested while the page is being loaded. From a user perspective, a session consists of visiting a number of pages in a certain order, usually by clicking on hyperlinks to navigate from one page to the next. Therefore we can simply filter the requests by extension. Depending on the structure and technology used by the website, we could for example retain the '.html', '.jsp' or '.pdf' pages but filter out the '.jpeg', '.gif' or '.css' files.

Listing Documents

After the filtering we can compile a list of all documents requested in the logfile. When pages are dynamically generated by the web server, it requires some consideration whether the same page with different parameters is considered as one document or as many different documents. This depends mostly on how strongly the dynamic content influences which pages can be reached from the dynamically generated page. If for example no links to other pages are generated dynamically then the parameters can often be ignored. For certain types of dynamic pages the content generated can depend heavily on session information that is not recorded in the logfiles and then the session of the user can not be reconstructed from the logfiles alone and the techniques used in this paper can not be used when only the logfiles are available.

In a similar way we can opt to group together some similar pages as one state in our model. This could be useful if we do not wish to go into detail about the navigation of users in one or more parts of the website, for example all pages concerning a special project. This is especially useful if this part of the website is only visited a small

number of times. Grouping these pages as one node means we can limit ourselves to studying the transitions in and out of this part of the website, and ignore all transitions internally in this group. The same is done by Li and Tian (2003). If part of the website uses frames, it can also be logical to group frames that are always loaded together, or to filter out frames that are often reloaded.

Sessions

From all the remaining requests we can now try to reconstruct the users' sessions, i.e. all the transitions from document to document by the users when surfing the website. For example the requests in Figure 3.1 together represent one successful session. Every session begins by a first request, which can be the site's main page, but can also be another page that was reached directly from an external referrer. Every session ends either by the users leaving the website (which we can only detect by an absence of requests from this user for a given amount of time) or by a request that results in error, after which the session is considered closed.

To group the requests in sessions, we need to group the requests by user and sort them into sessions. There are some problems with this approach. If different users are behind the same firewall or proxy server it might not be possible to discern these users from just the IP addresses. Also, when a user presses the 'back'-button in his web browser, the last page is not always reloaded and then no request is sent to the web server. The same is true if the user is looking at cached pages being stored either locally on the user's computer or at another location in the network. These transitions are then of course not logged in the server log. Probably, this is only a small part of the transitions, and can often be safely ignored.

To group requests into sessions we have to first define a session time-out. This means that we consider a request to be the start of a new session if there was no request from the same IP address for a given period of time, for example 30 minutes. A more detailed discussion of the influence of session time-out length can be found in the paper by Goševa-Popstojanova et al. (2006).

To reconstruct the browsing of a user we have two sources of information: the order of incoming requests (Goševa-Popstojanova et al. 2006) and the referrers logged for every request (Li and Tian 2003). For many users this information is consistent and the transitions can easily be deduced as is the case in Figure 3.1. However, when for example the user's browser does not indicate the referring URL, or if a user surfs the website in multiple browser windows at the same time, the referrer information can be inconsistent with the order of the requests. When constructing an algorithm to extract the sessions we have to take this problem into account.

Goševa-Popstojanova et al. (2006) define a failed session as any session containing at least one failed request (with return code 400 or higher). This means that a session is also considered failed if one image or style sheet can not be found. In this paper we employ a slightly different definition. First of all, we focus only on the pages and not their components, so we only consider a session failed if a complete page can not be returned. Secondly, we consider a session concluded after a failure, and therefore no

session can contain more than one failure.

It can also be discussed whether a return code of '401: Unauthorized' or '403: Forbidden' should be considered a failure. Rejecting a non-authorised user from seeing a certain webpage should be considered correct behaviour from the web server's point of view, but the user might experience this as a failure if he was expecting to have access to this webpage. The opposite of a failed session is a successfully concluded session, which is now defined as a series of requests from the same user which does not result in an error code and is concluded by a period of a certain length without requests from that user.

While compiling all the requests into sessions, we can filter out automatic reloads of the same page, since these actions do not reflect real transitions from a user point of view. Further it also makes sense to filter out all failed sessions consisting of only one request. Those failures are caused by a faulty link on external webpages or by mistyped paths. Since these faults are not under the control of the web server, they should not be included in the reliability of the web server. In the same way it makes sense to also exclude normal sessions that consist of only one request, since they do not actually contain any successful transitions through the website.

4 Markov Model

A Markov chain is a discrete-time stochastic process that has the Markov property (Winston 2003). Li and Tian (2003) conclude that Markov chains, extracted from logfiles can also be used as a good model for web usage. Empirical validation of the memoryless property typical for Markov chains shows that Markov chains are a practical and valid simplification.

By going through all the sessions, we can compose a large matrix listing the total number of transitions from every document to every other document. For most websites this matrix contains many zeroes, since most pages are only reachable from a few other pages. The matrix also records how often every document is the start or end of a session, either successful or leading to failure.

By then dividing every row by its sum, this matrix can easily be converted to estimates of the transition probabilities. This matrix of transition probabilities completely defines a Markov Model. With the help of this matrix the reliability of the website, defined as the possibility that a session terminates successfully (by a session time-out) and not in failure (when an error code is returned), can be calculated with the help of the formulas from Siegrist (1988). If all sessions could be perfectly reconstructed from the logfiles, the total reliability calculated from the Markov model would be equal to the percentage of successfully completed sessions.

5 Sensitivity Analysis

As we saw in the previous section, there is no need for a complicated Markov Model if we just want to calculate the reliability of the web server. However, if we want to analyse the sensitivity of the web server reliability to the changes in the usage, this section details how we can use the theory from (Weyns and Runeson 2007) to give us a good estimate. So the research question we are trying to answer now is how sensitive the reliability estimate is to the uncertainty or to changes in the transition probabilities between the different documents.

When we define an uncertainty on the transition probabilities we have to keep in mind that it is impossible for just one of the transition probabilities to change, since the sum of all transition probabilities out of one state always has to equal one. To solve this problem we use the mechanism of spread changes defined in (Weyns and Runeson 2007). This simply means that we allow the transition probabilities out of one state to change independently and then in the end divide them all by their total sum to make the sum equal to 1 again. This corresponds to modelling changes to the number of transitions instead of the derived transition probabilities.

We also need to define how we estimate the size of the uncertainty on each transition probability. In this article we discuss two options to estimate the uncertainty on the transition probabilities: based on the binomial distribution or based on an identical relative uncertainty on all transition probabilities.

5.1 Binomial Uncertainties

For this option we assume that the usage is the random result of a constant distribution of the transition probabilities. The binomial distribution tells us how we can estimate the transition probabilities from a sample, and how for a large sample the confidence interval can be approximated by a normal distribution.

For those cases where only a small sample is available, the transitions that only occurred a few times or never, the uncertainty on the estimate is much larger, and less symmetric than the simple normal approximation. This is most extreme for those transitions that did not occur during the whole period for which logfiles are available and for which the probability is therefore estimated at 0. Most of these transitions are indeed impossible transitions, but a small part might be from links that were never used during the period of the logfiles but could be used sometime later. For these cases the uncertainty is much harder to model and since all those transitions only contribute a small part to the overall reliability they are not worth the extra complexity from more advanced formulas such as can be found in (Brown et al. 2001). Nevertheless, we need to keep in mind that the estimate of the total uncertainty might be a slight underestimate if there are many pages for which the amount of requests is very low in the collected logfiles.

The result of applying the formulas of the binomial distribution is that the states that have been visited the least times, have the largest uncertainty. Since the behaviour

in these states is also less important for the total reliability, we can expect the total uncertainty on the reliability to be relatively low.

5.2 Relative Uncertainties

The second option for modelling the uncertainty on the transition probabilities takes into account the changing content and structure of the website. After an update of the website, the usage is likely to change abruptly. Then the real variation of the transition probabilities is actually much greater than predicted by the binomial distribution. In this case it is not possible to predict the change in the usage without more detailed knowledge about the planned updates, and even with this information it would still be hard to predict the exact changes in the usage.

If we do not know exactly how the website will be updated we have to resort to a more general model for the uncertainty. It is logical to assume that in general the larger transition probabilities have a larger absolute uncertainty. However, because there are so many small transition probabilities the uncertainty there can not just be ignored. Therefore we propose to use relative uncertainties. This means that we assume an equal relative uncertainty to all transition probabilities measured from the logfiles, for example 10% of their estimated value. Just like in the previous section, this simple assures us that the transition probabilities that were estimated at 0, remain 0 and that we can model the uncertainty as being symmetrical around the estimated value. Both of these conditions need to be fulfilled to allow for a statistical analysis as described in (Weyns and Runeson 2007)

If we would have access to a large quantity of log data and information about past updates, we could statistically analyse the effect of the updates on the usage of different parts of the webservers from the logfiles, but this is out of the scope of this paper.

6 Application to the Research Group Webpages

In this section we describe how we applied the method described above to the website of our research group, of which the latest version can be found on `http://serg.telecom.lth.se/`. The logfiles of two and a half months were collected and analysed. During those months no important changes were made to the website and we can expect the usage profile to be quite constant. The collected logfiles contain about two hundred thousand requests, and are together about 38 Mb in size.

6.1 Filtering

Filtering the traffic to remove the traffic caused by bots, reduces the amount of requests by 50%. The website under study contains relatively few images, compared to most commercial websites. Nevertheless, filtering out the images and other documents by file type we can still further reduce the amount of requests we need to analyse

by an extra 31%. Most of removed requests consisted of images, CSS-stylesheets and JavaScript source code files.

As discussed in Section 3.1, we can further remove all requests that represent a refresh of the previous request and all failures with an external referrer. This last number is small, but is very significant compared to the total number of failures. Finally we also removed all sessions of length one since they do not represent a real session.

After filtering we are left with about 5% of the requests which represent the real user transitions. When we group these requests into sessions in the next step we add another 2281 implicit transitions that do not show up as requests in the logfile because they represent the leaving of the website at the end of every successful session.

6.2 Sessions

For grouping the remaining request into sessions we chose to base ourselves mostly on the referrer information in every request. When the referrer was unknown or empty, the referrer was replaced with the target of the previous request when this last request occurred in the last half hour. Requests with an external referrer or with an empty referrer after a longer period of inactivity were recorded as the start of a new session. Further transitions to the success state were added for the last document before each period of inactivity.

Just the referrer information does not make a consistent Markov model. To make our model a valid model we have to complete the states for which no outgoing transitions were recorded, with a possible transition to success.

Of the requests that were not filtered out, 99.3 % did not indicate a failure, 66 requests had a return code of '403t' indicating a user trying to access a page he is not authorised to see, and only 5 requests were requests with an internal referrer and an return code of '404'. No other failure codes occurred. Only those last 5 requests were the result of broken links on the website, caused by only three different broken links. Broken links that lead out of the website under consideration can of course not be detected from the logfiles. Only internal broken links are taken into account here.

When fixing the broken links, we can easily remove them from the model and also use it to estimate the reliability of the website when the broken links have been repaired.

Because the collected logfiles start and end in the middle of the night on local time, there were very few requests from real users close to the start and end time of the logfile. Therefore we detected only one session of which we do not know if it continued beyond the end of the logfile, and there were no sessions starting in the first half hour after the start of the logfile.

We also opted to combine all the administrative pages of the website into one state, as described in Section 3.1. Further, we also combined all pdf-files in each directory into one state since they do not contain links to other pages and are usually quite similar.

6.3 Markov Model

Now we obtain a Markov model with 216 states: the 213 documents, a start state, a failure state and a success state. With this model we can very quickly calculate some statistics of the Markov chain we have used to model our system.

The most important statistic is of course the estimate of the reliability of the system. In our system this is calculated to 98.4%, which is of course lower than the request-based reliability. The remaining 1.6% probability of failure can be divided in a probability of 0.1% to end up in an 'error 404'-page, and 1.5% to end up blocked from an unauthorised page resulting in a '403'-error. The average session contains 4.73 transitions, and the session obtained by always following the maximum transition probability is exactly the session shown in Figure 3.1.

6.4 Sensitivity Analysis

With 13,000 transitions, we have 60 recorded transitions for every document, but only an average of 13 transitions for every of the 1042 different transitions that occurred at least once. There are many documents and even more transitions that have appeared only a few times or even only once. For those states and transitions it is impossible to make a good estimate of the transition probabilities. We can still use the binomial distribution to calculate an uncertainty on these transition probabilities, but the results are actually a slight underestimate.

Combining the formulas from (Brown et al. 2001) and (Weyns and Runeson 2007), we find an uncertainty of 9.4×10^{-4} on the reliability of 0.9837. Mathematically, this uncertainty represents the standard deviation that can be used to compute a confidence interval. This represents about 6% of the probability to end up in failure, and is therefore significant.

A second method to estimate the uncertainty is to consider a relative uncertainty of for example 10% on all the transition probabilities. This roughly represents the intuitive notion that the calculated usage profile is a good indication but is still a bit uncertain. With the formulas from (Weyns and Runeson 2007) this gives an uncertainty of 8.3×10^{-4}, or 5% of the probability to end up in failure.

When we would want to calculate the same result by a Monte Carlo simulation, this would require a substantial amount of calculation, and for systems with a few times as many states as the small website under consideration here, the simulation quickly becomes impractical. A 15 minute Matlab simulation on a normal desktop PC produced Figure 3.2. Each of the dots represents the reliability of a system of which all the transition probabilities have been randomly altered by using a normal distribution with standard deviation of 10% of their original size. The dashed line represents the predicted change in the reliability. The graph shows that the distribution of the simulated reliability conforms well to the predicted normal distribution with the standard deviation calculated above.

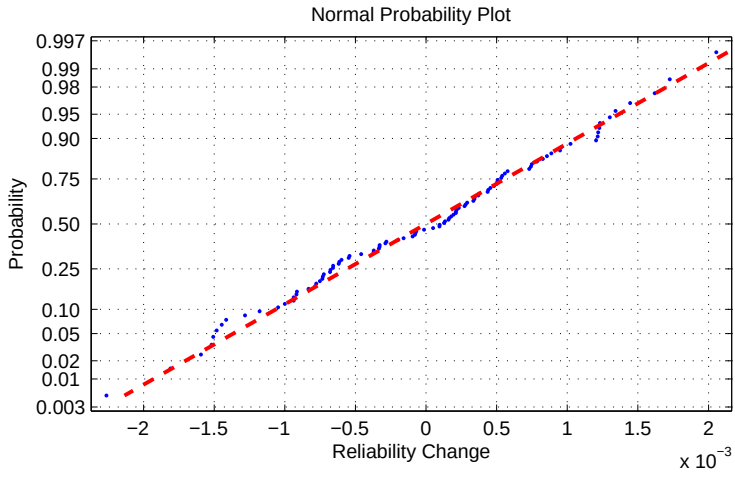

Figure 3.2: Predicted *(dashed line)* and simulated *(100 dots)* sensitivity to random, relative changes with $\sigma = 10\%$

7 Summary and Future Work

We have shown how the logfiles of a web server can be used to extract a usage profile from which we can estimate the reliability of the web server from a user's point of view. This can also be used to calculate the uncertainty on this reliability estimate. By applying the theory to the website of the research group, we have at the same time shown that the formulas from (Weyns and Runeson 2007) can be applied to a larger system in a useful way when usage statistics can be collected.

The disadvantage of this approach is that it requires a lot of pre-processing of the data, before a usage profile can be extracted. At the moment there is little tool support for this. Further, the approach is limited by the data logged in the logfiles, we can, for example, not detect any broken links leading out of the website.

Further work on this approach could include improving tool support, applying it to more and larger websites and exploring the influence of incorporating more advanced formulas for the confidence intervals from the binomial distribution from (Brown et al. 2001). An important extension could also be to combine this approach with more detailed predictions of how the usage of the website is expected to change in the near future. These predictions could then be used to predict the reliability of the website after changes to the website or when the usage is expected to change substantially.

Bibliography

L. D. Brown, T. Cai, and A. DasGupta. Interval Estimation for a Binomial Proportion. *Statistical Science*, 16(2):101–117, 2001.

K. Goševa-Popstojanova, A. D. Singh, S. Mazimdar, and F. Li. Empirical Characterization of Session–Based Workload and Reliability for Web Servers. *Empirical Software Engineering*, 11(1):71–117, 2006.

Z. Li and J. Tian. Testing the suitability of Markov chains as Web usage models. In *Proceedings of the 27th Annual International Computer Software and Applications Conference*, pages 356–361, 2003.

L. Ma and J. Tian. Web error classification and analysis for reliability improvement. *Journal of Systems and Software*, 80(6):795–804, 2007.

J. C. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-lee. Hypertext Transfer Protocol – HTTP/1.1, Request for Comments 2616, Network Working Group, 1999. URL: `http://rfc.net/rfc2616.html` (Last accessed November 2007).

National Research Council. The Internet Under Crisis Conditions: Learning from September 11. Technical report, 2003. URL: `http://www.nap.edu/catalog/10569.html` (Last accessed November 2007).

S. Sampath, V. Mihaylov, A. Souter, and L. Pollock. A scalable approach to user-session based testing of web applications through concept analysis. In *Proceedings of the 19th International Conference on Automated Software Engineering*, pages 132–141, 2004.

K. Siegrist. Reliability of systems with markov transfer of control. *IEEE Transactions on Software Engineering*, 14(7):1049–1053, 1988.

P.-N. Tan and V. Kumar. Discovery of Web Robot Sessions Based on their Navigational Patterns. *Data Mining and Knowledge Discovery*, 6(1):9–35, 2002.

K. Weyns and P. Runeson. Sensitivity of Software System Reliability to Usage Profile Changes. In *Proceedings of the 2007 ACM Symposium on Applied Computing*, pages 1440–1444, 2007.

W. L. Winston. *Operations Research: Applications and Algorithms*. Duxbury Press, 2003.

Part II

Case Studies on the Dependability of IT systems in Emergency Management at Swedish Municipalities

Paper IV

Dependability of IT Systems in Emergency Management at Swedish Municipalities

Kim Weyns and Martin Höst

Presented at the 7th Conference on Software Engineering Research and Practice in Sweden (SERPS'07), October 2007, Göteborg

ABSTRACT

In recent years municipalities have become more and more dependent on IT systems for their responsibilities in a crisis situation. To avoid unexpected problems with IT systems in the aftermath of a crisis it is important that these risks are identified before a crisis occurs and that measures can be taken to reduce the dependence on systems that could be unreliable. This report describes the results of two case studies exploring how Swedish municipalities incorporate IT systems in their emergency planning. Interviews were conducted with both emergency managers and IT personnel, and data from the interviews is combined with data from a large survey. The study focuses especially on how different actors within a municipality cooperate to analyse the risks of depending on IT systems in critical situations. The study shows that today there is much room for improvement, especially in the communication between IT personnel and emergency managers.

1 Introduction

Swedish municipalities have an important active role in crisis relief. To prepare for these crisis situations, each municipality employs a number of emergency managers. Their main task consists of conducting vulnerability analyses and to use the results of these analyses to improve the municipality's ability to offer crisis relief in the aftermath of all kinds of crises while at the same time keeping their most critical services operational.

In recent years municipalities have come to depend more on IT systems for all their every day workings. For communications municipalities of course depend on landline telephone networks, mobile phone networks, webservers, email servers, etc. Other important systems are used for patient administration in health care and social care, school administration or city planning. Further, a lot of critical information is no longer stored on paper, but is only available in electronic format, either locally or even on a server located far away.

In the same way, municipalities now depend on all kinds of IT systems for their responsibilities in crisis situations. Under normal conditions, an occasional unavailability of most IT systems is fully acceptable, but during crisis situations where time is a critical factor in the relief efforts, any unexpected unavailability can have disastrous consequences. Therefore it is important that IT systems are an integral part of all major vulnerability analyses conducted. These vulnerability analyses are needed to combine information about the dependability of the different IT systems with information about how critical the systems are in different situations. A high dependency is only acceptable on systems which are highly trustworthy. Less reliable systems can also be part of emergency plans, but only if alternative solutions are available, reducing the criticality of the systems.

This vulnerability analysis is not always as straightforward as it may seem. The main complicating factor is that the information necessary is spread about over many people. Conducting the vulnerability analysis is usually the task of the emergency managers, who also work with the emergency scenarios the municipality is preparing for. Detailed information about the reliability of the IT systems is often only known to the manufacturers of the systems and the IT personnel responsible for the maintenance of the systems. In many cases this can even be external service providers that provide the support. In the worst case, when no failure statistics are systematically collected and little acceptance testing was done, no dependability information is available. Further, because reliability information is often expressed in a technical way, it can be hard to incorporate into vulnerability analyses by emergency managers without advanced knowledge about software reliability.

Detailed information on how critical certain IT systems are and how they are used in different situations is usually only available to the actual users who depend on the IT systems, and who often do not think about the crisis situations that could occur. They also often base their view of the reliability of the system completely on their own limited past experience with the system.

A second difficulty with this vulnerability analysis is that IT systems can exhibit very complex failure behaviour. A system that has worked perfectly for years, during normal operating conditions, is in no way guaranteed to work correctly in special usage scenarios. Unfortunately, these special usage scenarios are exactly what might occur during crisis situations, which are by definition very rare events. Because of the complexity of most IT systems, it can be very hard to predict which combination of environmental and usage changes can have a large negative impact on system reliability. Even if we manage to understand this relationship between a changing usage and the dependability much better, for example through a detailed study of the sensitivity of the reliability to usage changes as described by Weyns and Runeson (2007) and by Goševa-Popstojanova and Kamavaram (2004), it would still be difficult to predict which changes in usage we could expect during certain crisis situations because also the interactions between IT systems and users can be very complex.

A third important complicating factor is that IT systems tend to change very quickly. New systems are added regularly and old systems are almost constantly being updated. At the same time IT systems are also used for more and more new functions all the time. Good risk management would require an updated vulnerability analysis with each important change in the systems or in the way they are used. Practically this is usually impossible, but much improvement in this area is possible.

In this article we present the results of two case studies at Swedish municipalities about how they include IT systems in their emergency planning and vulnerability analyses. Additionally some results from a survey conducted among the IT safety responsible at 230 governmental actors in Sweden are presented. Section 2 presents some general background information about the role of Swedish municipalities in the Swedish emergency management system. Section 3 gives an overview of some related publications. In Section 4 the research methodology used to conduct the case studies and the survey is discussed. Section 5 discusses the main finding of the study. Some threats to the validity of the study are discussed in Section 6. Finally the main conclusions of the study and a discussion of possible future work can be found in Section 7.

2 Background

2.1 Dependability

In this paper we will use terms such as software dependability, reliability and security as they are defined in the work from Avizienis et al. (2004). This means that dependability is defined as the most general concept, encompassing the more limited concepts of reliability, availability, safety and security. Reliability is mostly concerned with how often failures occur in the system. Availability takes into account how long the system is not functioning when failures occur. Safety is concerned with the absence of failures causing catastrophic consequences for its users and the environment, while security on the other hand, describes how sensitive the system is to external

> ”Crises are events that disrupt the functioning of society or jeopardise the conditions that govern the life of the population. They include serious crises in times of peace as well as war. Such situations demand good emergency management if they are not to undermine confidence in the Government and authorities and potentially threaten the national security and democracy of Sweden.”(SEMA)

Figure 4.1: SEMA's definition of a crisis

threats. Dependability takes into account all these aspects and corresponds best to the intuitive notion of how much a system can safely be depended upon by its users.

2.2 Emergency Management in Sweden

Swedish emergency management (KBM 2005) is mainly based on the principle of responsibility, which means that the in emergency conditions the responsibilities for everyday matters should still be with those governmental actors that are also responsible for these matters in normal conditions. Through the principles of proximity and geographic area responsibility, emergency management is in the first place a responsibility of the local governments. Practically, this means that municipalities are the central actor in crisis planning and crisis relief. Only with major crises that affect many municipalities the regional governments are directly involved in an operative role.

For their emergency planning, Swedish municipalities receive support from the regional governments and the Swedish Emergency Management Agency (SEMA). The regional councils have the responsibility to coordinate the emergency management at a regional level and to systematically review the emergency plans of the municipalities in each region and to report on this to SEMA. SEMA itself assists the regions and municipalities by supporting them in emergency planning and by providing information and guidelines. Unlike in many other countries, SEMA does not have an operative role in crisis relief.

SEMA defines a crisis as in Figure 2.2. Informally it could be stated that a crisis is when a combination of events, e.g. accidents or sabotage, result in a situation that negatively affects society in a way that hinders vital society functions. Examples of crises that are included in this definition might be terror attacks, storms, tsunamis, murders etc.

2.3 Swedish Municipalities

Sweden is divided into 290 municipalities (SALAR 2005). The population of Swedish municipality range from under 2.500 in Bjurholm to 770.000 in Stockholm. Their geographic area ranges from 9 km^2 for Sundbyberg to 19.447 square km^2 for Kiruna.

The municipalities are responsible for the matters directly relating to their inhabitants and their geographic area. This means that their main responsibilities include education, child and elderly care, street maintenance and emergency management. Therefore Swedish municipalities are both service providers (for social care and education) and supervisory authorities (for environmental issues for example).

There is no standard organizational structure shared by all municipalities, but some common factors can be seen in nearly all municipalities. The main regulation governing the workings of a municipality is called the 'Local Government Act' (KL 1991:900). The activities of a municipality are lead by a municipal executive committee, appointed by the elected representatives. The daily work is lead by a municipal director, who reports to the municipal executive committee.

The main activities of a municipality are usually divided over a number of administrative units, each responsible for one or more of working areas of a municipality (social service, social care, city planning, environmental issues, emergency services, culture, etc.). These activities are all external services the municipality offers to the general public. To support all these external services there is a need for a number of supporting activities, also called internal services, such as economy, technical support, housing or IT.

These internal services can be centralised for the whole municipality or divided over the different administrative units, depending on the organisational structure of the municipality. Many municipalities have recently brought their IT personnel into one central IT unit that offers IT services to all administrative units. This allows for a more efficient use of IT resources than before when many of the administrative units had their own separate IT personnel. We further discuss the consequences of this reorganisation in section 5.1

As said before, one of the responsibilities of a municipality is emergency management. Therefore most municipalities have one or more emergency managers who are often part of the fire department. Their responsibilities usually range from making emergency plans and conducting vulnerability analyses to organizing the information flow under an actual crisis.

3 Related Work

3.1 Emergency Management

In many countries emergency planning is coordinated on a national level by a federal government agency such as the United States Federal Emergency Management Agency, Emergency Management Australia, Public Safety Canada or the Russian Ministry of Extraordinary Situations. Although the exact roles of these agencies can differ from country to country, they all support local governments in their emergency management. Because emergency management is handled differently in different countries, most countries published their own vulnerability analysis methods for use at the local level. In Sweden most publications on this topic are published in cooperation

with SEMA. A good overview of Swedish emergency planning at the municipal level can be found in (Hallin et al. 2004) and (KBM, 2004). Hallin et al. (2004) also describe a scenario based method called Municipal Vulnerability Analysis, MVA.

In the private sector emergency management is usually called business continuity management. An important difference between the public and the private sector in this field is that governmental actors often have an important, active role in crisis relief and need to prepare to offer special services in the aftermath of a crisis. Business continuity management is concerned with keeping the level of service at a normal level in crisis conditions, or degrading the level of service gracefully to an acceptable level. For a governmental actor on the other hand, the unavailability of its services might be fully acceptable on any normal day, but can be critical in crisis situations. This special role in crisis relief role poses completely different demands on their emergency management procedures than those used in the private sector.

3.2 IT Management

A number of international best practice frameworks and standards have been published to help organisations obtain a higher dependability of their IT services and systems, among those (ITIL), (COBIT) and (ISO-IEC 17799). These frameworks are much more suited to be used by large corporations with very large IT resources. For small municipalities these frameworks are too large to be of any practical use.

For this reason, SEMA published the BITS, Basic Level for IT Security, handbook (BITS). BITS is meant to give Swedish authorities a practical overview of the main administrative measures that can be taken to achieve a minimum level of IT dependability. BITS is based on international standards such as ISO-IEC 17799, but BITS is much more suited for small public actors. BITS is also accompanied by BITS Plus, a web based planning tool that can be used to coordinate the work with the BITS standard. The main disadvantage with using BITS for achieving a higher dependability is that it focuses mainly on security and a lot less on reliability and safety. Most of the chapters focus only on confidentiality and integrity, without discussing other than malicious threats to the dependability of the system. Secondly, BITS also focuses mostly on the technical system level which makes it easy to loose track of the organisational level and of how critical the systems actually are for the organisation in different situations. This was also remarked in the survey described in Section 4.2. Overall this makes BITS a good tool for systematic work with IT security matters, but BITS is only part of the solution needed for evaluating the dependability of IT systems during a crisis.

Internationally, more and more research is being done on special systems that can be used in crisis relief, but many of these systems are only in the development phase. The near future will almost certainly see a serious rise in the number of IT systems used in crisis situations. So far most of these systems are only considered as an extra tool in the aftermath of a crisis, but as these tools become more common, we come to depend on them more and more. Just like when people start using a mobile phone, they first see it as a tool that just makes some things a bit easier. However, after using a

mobile phone for some time, they can no longer imagine how they could ever manage without a mobile phone. Therefore, extra caution is warranted when these crisis relief systems are ready to be used in real emergency situations. To be able to use these systems efficiently, and to be able to evaluate the dependency on these systems, it is even more important to fully integrate these IT systems into emergency management and include them in the vulnerability analyses that are conducted.

4 Research Methodology

The research in this report combines results from three different sources: an elaborate literature study, data collected from case studies at two Swedish municipalities and the data of a survey conducted by SEMA among all Swedish municipalities and a series of other governmental actors. The two case studies are described in Section 4.1 and the survey is elaborated upon in Section 4.2.

4.1 Case Studies

The main part of this research was conducted in two case studies at two different Swedish municipalities. The municipalities were specially selected because they had shown an interest in the topic of IT systems in emergency management in previous contacts with SEMA or with other members of our research project.

Municipality A is a large Swedish municipality consisting of a major Swedish harbour city and the surrounding urban areas. Municipality B on the other hand is a small municipality consisting of two suburbs of a large Swedish city. The two municipalities are substantially different in many important ways. Municipality A has roughly 6 times more inhabitants and also employs about 7 times more people. Also from a vulnerability perspective there are large differences. Municipality A houses a lot of industry and is an important national hub for the transport of dangerous goods. During the last years the municipality has gone through some major emergency situations of different types. Municipality B on the other hand has a much lower risk profile and has not experienced any major emergency situations in the last 15 years.

To understand how these municipalities assess the dependability of their IT systems in emergency situations, a series of interviews were conducted with emergency managers and IT personnel at both municipalities. Further a number of documents concerning IT strategies, organisational structures and vulnerability analysis were also collected and studied, both before and after the interviews. Because the nature of the research was strongly explorative, each consecutive interview or document was studied immediately and this information was used to improve the preparations for the next interviews. The disadvantage with this method is that it might introduce a bias in the next interviews, but it was considered that the advantage of a more informed preparation for further interviews outweighed this disadvantage, especially since some of this researcher bias is inevitable when all interviews are conducted by the same researchers.

The interviews were conducted as open interviews (Robson 2002), with a lot of free-

dom for the respondents to give their view on the issues at hand. All the interviews used the same basic list of open questions, but they were only used to make sure the interviews covered all the necessary topics, not to decide the order of the topics. Because different municipalities have such different ways of working, it was not possible to compile a list of very specific questions that could be used for all the interviews. Often it was necessary for the interviewees to first explain a number of other aspects before they could completely answer a certain question. The advantage of this freedom in the interview is that the respondents had the freedom to stress the parts they see as the most important. The main disadvantage is that the analysis of the interviews becomes harder because there is not standard structure.

For the first municipality, interviews were conducted with two emergency managers and one former IT manager, currently working at the social care department as project manager, specialised in IT projects. At municipality B, interviews were conducted with one emergency manager responsible for IT safety and one IT technician. At both municipalities the emergency managers were interviewed first, since they are easier to contact for an outsider. They were then asked to provide contact information to suitable contacts in the IT department.

For the analysis, all interviews except the first one were recorded and transcribed in full. During the transcription they were also translated from Swedish to English to facilitate the analysis. As recommended by Robson (2002), a number of coding categories were used to reduce the amount of data to be studied. For the coding two independent researchers went through all the transcribed text and coded all passage that related to one or more of the following categories and subcategories:

- Organisation
  - Organisational structure
  - Responsibility for the IT systems
  - Organisational changes
- Risk analysis
  - Risk analysis activities
  - Identification of critical systems
  - Prioritisation of IT support
- Communication with the IT personnel
  - Ways of communication
  - Driving force for communication
- Service level agreements, SLA
  - SLA form
  - SLA content

- Practical examples
  - Past problems
  - Frustrations
  - Implemented solutions and practices.

These final categories are the result of a stepwise improvement from an initial set of categories based on the main concepts in the research. The coding helps us to identify statements that logically belong together but are spread out over the text. The coding was not a goal on its own, but an analysis tool and therefore the categories were not defined too strictly beforehand and it was left to the researchers doing the coding to fine tune the categories.

The first category collects statements about the personnel involved in evaluating the dependability of the IT systems. The focus of this category is on how the responsibilities are divided between the different people involved. The second category contains all data about how vulnerability analysis is performed at the municipalities under study, with special focus on how the IT systems are analysed. The third category collects the information about how and when the IT unit communicates with the rest of the municipality's personnel. The fourth top category is about service level agreements in the very broad sense, so everything about the level of services expected from the IT systems at the municipalities, and how this is specified or agreed upon. The last category collects all the practical examples that were discussed during the interviews that were most illustrative for the issues discussed in this report.

After two separate researchers, i.e. the authors of this paper, marked the interviews according to this categorisation, their lists were merged and the excerpts in every category and subcategory were analysed. Since the interviews often returned to the same topic, and because different people in the same organisation were interviewed, a triangulation can be done to check the consistency of the interviewees' answers.

For the analysis both within and across the two municipalities the technique of explanation building as described by Yin (2003) was used. Special attention was given to those issues where the respondents disagreed or gave conflicting answers. More details on how the conclusions were reached from the data can be found in the discussion of the study's findings in Section 5.

4.2 Survey by SEMA

In May 2005, SEMA conducted a survey among 368 IT security managers at Swedish municipalities, regional governments and different public authorities. A first analysis of the 230 answers to the survey they received was published shortly afterwards (Kalmelid and Gustavsson 2005).

The survey consisted of between 14 and 30 questions, depending on the chosen alternatives. The majority of the questions were multiple choice questions where respondents were asked to rate something on a scale from 1 to 5. A substantial part of the

questions were open questions that gave the respondents the chance to explain their answers in more detail.

The goal of the survey was to assess the capabilities of different governmental actors in the field of IT security. Within IT security the survey focused mostly on the methods and standards used and how SEMA's support towards the governmental actors could be improved. The respondents were also asked to make an assessment of the maturity of their organisation and different members of their organisation in IT security.

For this report we had access to all the raw data from the survey, but all names were removed for integrity reasons. It was still possible to determine if a series of answers came from a municipality or a regional government, but answers could no longer be connected to a specific public actor. The answers to the multiple choice questions were mostly used to see how common the use of the different methods and standards is. The answers to the open questions were analysed in a similar way as the interviews in the case studies. The main conclusion and a graphical analysis of the multiple choice questions can be found in the survey report (Kalmelid and Gustavsson 2005).

When considering the validity of the data collected from the survey we need to keep in mind that the answers only reflect the view of the IT security responsible at each public authority. To get a complete picture other roles in the municipality should be included in the survey too. Secondly, it might be possible that those public actors that have the lowest level of maturity in IT security did not care to answer the survey, and this would make that the results can not be generalized blindly. Further it is important to see that the focus of the survey and the case studies is slightly different. The survey focussed on security, while the case studies were concerned with dependability.

5 Findings

This section contains the main findings from the case studies and the survey. Each of the next sections discusses the conclusions that can be drawn from the excerpts that were coded in to the corresponding categories and subcategories. Therefore the following sections follow roughly the structure of the categories listed in Section 4.1, though the order has been changed to facilitate the discourse.

5.1 Organisation of IT Services

System Responsibility

In both municipalities that participated in the study there is a central IT unit responsible for the maintenance of the IT systems. For some systems, the maintenance is done with the help of suppliers or external consultants. When it comes to the final responsibility for the system, both municipalities make a distinction between those systems that are common for the whole municipality and those that are specific for one department. The former systems such as the email system, the network or the operating systems are the direct responsibility of the IT department. The latter systems

such as the economy system or systems used in social care are the responsibility of the specific departments. This responsibility means they decide about the acquisition, the updates and the evaluation of the systems. The maintenance for both types of systems can be performed by either the IT department or by external consultants, for example from the supplier of the system. The contracts with the supplier can be signed with or without some involvement of the IT department.

The main advantage of this approach is that the main responsibility for all the systems lies with those who have the most knowledge of the application area of the system. This approach also has a number of problems, especially in the cooperation between the IT department and the people responsible for the systems owned by the different departments. Because they are in different departments with different goals, there is often a conflict relationship between them prohibiting a good cooperation and exchange of necessary information.

A first problem lies in the evaluation of the dependability of the systems. Since the IT department is responsible for the maintenance they are contacted in case of any problems, but it is not their responsibility to collect failure statistics, as expressed in Quote 1. The IT personnel has the ungrateful role of having to maintain these systems while they can not directly influence their administration. The responsible of the system on the other hand, is then not even notified of all the problems, and can not get a full picture of the dependability of the system. In municipality A, the IT department has a help desk that coordinates the maintenance work of the IT department. In municipality B, users contact one of the employees of the IT department directly on their mobile phone, making it even harder to collect failure statistics. Further, concerning the service that is outsourced to external suppliers, some failures are reported directly to the supplier, while others are reported to the supplier through the IT department.

> **Quote 1**
> *We don't do any organised collection of statistics now, we just try to solve the problems that pop up.* – IT TECHNICIAN, MUNICIPALITY B

A second problem is that most of the systems owned by the various departments are dependent on the operating systems and the network administrated by the IT department. Since both groups have the individual responsibility to decide about major updates to their systems, this can create problems when these are not communicated well in advance.

A third problem is that in this organisational structure the IT-departments do not have any own technical personnel that can advice them on the technical details that are involved in the administration of the systems they are responsible for. To be able to take full responsibility for the systems not only a good understanding of the purpose of the systems but also a good technical understanding of the workings of the system is necessary. This can lead to responsibilities implicitly being shifted to the IT department where they do not belong, just because the different departments do not

immediately know how to deal with them. This is for example complained about in the survey as can be seen in Quote 2.

> **Quote 2**
> *To define the limits of their area of responsibility to make sure that the responsibility is where it should be. This is necessary to avoid that the stress lies on the technology in stead of the processes. We are not good enough at explaining that there are some parts where the different departments must take responsibility. Today it is automatically the IT unit that must take responsibility for IT matters for which no-one else takes responsibility. This is not good.* – SURVEY ANSWER TO THE QUESTION: WHAT DO YOU THINK THE IT PERSONNEL COULD GET BETTER AT CONCERNING IT DEPENDABILITY?

Another common problem with the organisation of the IT department is that in old organisational structures IT is still considered to be a part of the economy department and the CIO, Chief Information Officer, or a similar function, still reports to the head of the economy department, and not to the director of the municipality directly. This is also referred to as an important inhibitor to a good cooperation between the IT department and the rest of an organisation in the IT governance literature, for example in the work by Luftman (2003). This problem was also visible at municipality B and was complained about in the survey.

In municipality B, IT safety is a responsibility of one of the emergency managers at the municipality. The advantage of this role is that he can lift these safety issues immediately to the highest levels in the municipality where the legal responsibility for all safety matters in the municipality lies. On the other hand, the danger is that the IT department feels relieved of all safety responsibilities although their expertise is indispensable for evaluating this safety.

Internal Communication

An often recurring complaint, in the case studies and the survey, is a lack of real understanding between the IT department and the users. Users complain that the IT personnel does not understand what they expect of their systems, as for example in Quote 3. The IT personnel on the other hand complains that the users do not understand the risks involved with IT systems, especially concerning security.

> **Quote 3**
> *We have generators and we can provide backup power to our IT systems very long. Quality of the IT systems is harder. We have discussed this a lot. Also with our IT technicians, but they focus often on the wrong things.* – EMERGENCY MANAGER, MUNICIPALITY B

This lack of understanding is a consequence of the communication problems between both parties. Both municipalities under study lacked a forum where the IT department and the users could discuss important IT issues together, as discussed in Quote 4. For some major discussions working groups are created that include representatives of suppliers, users and the IT department, but this is done too seldom. In the worst case the only communication occurs when a failure of a software system occurs and the IT department has to be notified to fix the problem.

> **Quote 4**
> *They always want to buy a new server for every application, but that is not always necessary. But it is not us who decides, we just hopefully get asked, though often too late.*
> – IT TECHNICIAN, MUNICIPALITY B

For example, when it comes to communicating about major updates to the systems, under the responsibility of either the IT department or the other departments, both municipalities admitted that they had encountered problems in the past. All people involved knew that the best way would be to discuss any major updates with all parties involved before the decision to update is made final, but in practice many decision where made unilaterally and sometimes the other parties were not even notified in advance of the update.

Another common complaint about the communication between users and the IT department is that the communication from the IT department is too technical. Outside the IT department there is not enough technical knowledge to understand the technical details of the system, while the IT department does not manage to communicate their message without resorting to technical details. This adds to the frustration of parties, and results in the IT department not being consulted as often as necessary for important decisions.

Service Level Agreements

Both municipalities in the study have some service level agreements, SLAs, with their external suppliers but have no service level agreements at all with their own IT department. In a small municipality it would probably be too tedious to write formal agreements that should be considered as binding contracts. Nevertheless, some written communication where users and the IT department discuss the level of service, could bring clear advantages to both parties. For example in municipality B, the IT department tries to always have some IT personnel reachable to provide service, even in weekends and at night in case there is a need for urgent IT support for critical systems. This level of service is in no way guaranteed to the rest of the municipality, but is just done because the IT department considers it reasonable.

Without SLAs the IT department is expected to deliver services at best effort, but without any specifications what level this is. In this situation, all failures are considered as faults of the IT department to deliver satisfactory service. Further there are no

written agreement as to which systems should have a high availability, and the IT personnel estimates from experience which systems are most critical to prioritise their work. Service level agreements would give the IT department a stronger position when asking for resources to deliver a necessary level of service and at the same time protect them from user expecting an impossibly high level of service, as expressed in Quote 5.

Quote 5
There are 1000 reasons for having a service level agreement, but the one reason for not having it is that without one, the IT department is not obligated to anything. They do not see that it could also be a defence for them that they can not be blamed for not delivering something they before clearly stated they could not deliver. – PROJECT MANAGER, MUNICIPALITY A

The advantage of SLAs for the users is that they know what to expect, and what not to expect, from their IT systems. This way they can avoid both depending on unreliable systems and investing in unnecessary backup solutions for sufficiently reliable systems. This problem is expressed in Quote 6 from a project manager at municipality A.

Quote 6
If the IT department can explicitly state that they can not give us any guarantees, we have good reason to invest some extra millions on this side to secure our systems. But now we have no arguments to justify this cost here. – PROJECT MANAGER, MUNICIPALITY A

Even the service level agreements with external suppliers are often not well planned and not adapted to the level of quality actually demanded by the users of the systems. For example at municipality B, the maintenance contract with their supplier of routers guaranteed on-site service within 8 hours. This number was agreed upon many years ago, and nobody seems to know exactly why it once was set at 8 hours. The importance of the internal network for the daily operations at the municipality has definitely increased drastically since this decision was taken. This example shows there are no routines in place to regularly re-evaluate important service level agreements.

Service level agreements are closely connected to measurements. The writing of service level agreements forces an organisation to think about how the quality of its IT systems can be measured. Just as both municipalities lack service level agreement for most of their systems, they also lack the possibility to measure the quality of their IT systems. Access to such measurements would give them a possibility to concentrate their resources better to improve the weakest links in their critical systems.

5.2 Emergency Management

Every municipality has a number of emergency managers responsible for preparing the municipality for possible crisis situations. An important part of this task is to help all the departments in the municipality to conduct risk and vulnerability analyses and to produce emergency plans. The risk analyses can only be conducted by the personnel of each department, because they are the only ones that have the necessary knowledge about how emergency situations could influence their work. The emergency managers help them in this task by instructing them in the methods that can be used, and reminding them to keep their emergency plans updated.

The most commonly used methods for risk and vulnerability analysis are scenario-based, as for example the method developed by Hallin et al. (2004). Most municipalities also organise regular, scenario-based emergency exercises to test their emergency planning. The emergency management of a municipality often results in a number of simple measures that can be taken to seriously reduce the probability or effect of possible crisis situations.

The emergency managers are also responsible for planning the specific responsibilities of the municipalities in crisis relief and information spreading during a crisis. All municipalities are required to have a crisis central that can be used to coordinate the relief effort during and in the immediate aftermath of a crisis. SEMA also assists municipalities in setting up such a crisis central and analysing which facilities are required. IT systems, and especially communication systems, are an important part of the equipment available in a crisis central.

Although IT systems can play an important role in the aftermath of a crisis, they are seldom included in the emergency plans and risk analyses that are conducted. Emergency managers would like to include these systems, but in practice they do not manage to cooperate with the IT department to do so. In municipality A, the emergency management of the social care department is planned to be completely independent of IT systems. This means, for example, that all critical information is printed out on a very regular basis and communication plans are ready that do not rely on modern technology. As the project manager explained, this is a safe solution, since it means they are prepared for a complete failure of all IT systems, but it is also a serious overhead cost that is only necessary because they do not manage to analyse the risks of depending on their IT systems. If they would manage to include the IT systems in the risk analyses, they would be able to evaluate which systems are reliable enough to depend upon in different emergency situations, and they could safely reduce this overhead cost. Because the IT systems are not part of the emergency plans, they can also not be used as efficiently in a crisis if they turn out to be reliable after all.

In municipality B, a crisis central was installed with the help of SEMA and a number of external consultants. Although this room contains a number of computers and network connections, the IT department was not involved in the development of this room. The systems in this room are meant to be used in crisis situations and have redundant phone and Internet connections. The IT department also maintains the

systems in this room, but they have no responsibility for the reliability of these systems and are not involved in any strategic planning of how the systems in this room should be updated or replaced.

When the IT department is not involved in emergency planning, as expressed in Quotes 7 and 8, they are also not aware of which systems are critical during different crisis situations and they can not correctly prioritise their maintenance work without receiving specific instructions during a crisis. This also means that IT systems are seldom involved in emergency exercises. Useful lessons could be learned from exercises such as regularly trying to restore a system from backup, or measuring the behaviour of the network when one or more routers are disabled. When this kind of statistics is available it can be taken into account in the emergency planning.

Quote 7
We are not involved in making emergency plans. It's not something we think about. – IT TECHNICIAN, MUNICIPALITY B

Quote 8
I don't know what the rules are for prioritised service in an emergency. Nobody said that one computer is more important than the others. – IT TECHNICIAN, MUNICIPALITY B

5.3 Common Problems

In this section we summarize the main problems the studied municipalities experienced when trying to integrate their IT systems in their emergency planning.

A first recurring problem is the lack of good supporting tools or standards. BITS (2003), the brochure with guidelines published by SEMA, is used by 75% of the municipalities that answered the survey, but BITS Plus, the tool that was added more recently, was used by only 28%. BITS is more focused on security than reliability, and the focus is therefore more on the systems as separate units, and not on how the systems fit in to the overall activities of the municipality, as also remarked in the survey as in Quote 9. For this reason, BITS is not ideal for a complete dependability analysis, and might even lead to some aspects being forgotten when it is not complemented with other risk analysis tools or methods that incorporate the IT systems. The international standards and best practice frameworks such as ITIL and COBIT discussed in Section 3.2 are too large and too much focused on companies to be very useful to most municipalities.

Quote 9

The main disadvantage of BITS is that it uses an object-oriented model for IT dependability, instead of a process-oriented model. This means it sees IT systems as isolated objects, in stead of starting from the information processes that are provided or supported by the system. – SURVEY ANSWER TO THE QUESTION: WHAT DO YOU THINK COULD BE IMPROVED ABOUT BITS?

For this reason, municipality B has started developing their own risk management tool, with special focus on following up the whole process from identification of possible risks to mitigation. When the system is completed, it is meant to be used by all departments in the municipality. At the time of this research, the systems was however only just being deployed and was not used for documenting IT risks yet.

A second major problem that was observed at both the municipalities was the problem with defining who is responsible for evaluating the dependability of the IT systems in crisis situations. This task requires the cooperation between the emergency managers, the IT department and the department owning the system. In practice, because of the communication problems discussed before, this can lead to this issue being overlooked when nobody takes the responsibility to organize a working group to tackle this problem. Especially if the IT department is not involved in the strategical discussions about the IT systems, they limit themselves to the daily maintenance of the systems and only perform technical long-term improvements when explicitly asked. This can for example be observed in Quote 10

Quote 10

– Interviewer: Computers have become more critical in the last years. Did you recently re-evaluate the 4 hour service agreement with your network supplier?
– No, this is something the users of the applications should worry about, not us. We only have a responsibility for the maintenance of our systems: the network, the mail servers, and file servers. – IT TECHNICIAN, MUNICIPALITY B

Another problem is the users' and emergency managers' limited understanding of the dependability issues of IT systems. Especially concerning security, as shown clearly in the survey, the IT department often complains about the negligence of the users. Also concerning the reliability, the users do not have enough technical knowledge to understand the IT systems. When they want to conduct a risk analysis of the IT systems they need this technical knowledge to be able to understand all the threats to the reliability of the system, their probability and possible consequences. Often it is assumed that the IT systems can be depended upon in a crisis, even if there is no evidence of their reliability.

Finally, a typical problem with IT systems is their fast evolution. New IT systems are

installed every year and updates are done even more regularly. Adding new systems or new functionality to old systems changes both the reliability of the system and the dependence on the system. When the municipalities already have some risk analyses of their IT systems, they do not manage to keep these analyses updated to reflect the latest functionality of the IT systems. This is especially important since the dependence on the IT systems is increasing continuously. At first, after a new system has been installed, the system is usually only considered an extra asset that could be useful in a crisis situation, even if it not critically necessary because the old alternatives are still available. At this time the dependability of the system is not critical, but when the user get more used to having the new system around, the alternative systems are neglected and the new systems can get more and more critical. When these changes occur gradually, they are sometimes only noticed too late and systems can become critical without their dependability ever having been seriously evaluated.

6 Validity Discussion

A number of possible threats for the validity can be identified for this study. Concerning external validity it is important to understand that the results of the case studies can not be generalized in the same way as the results of the survey.

The majority of the 290 Swedish municipalities participated in the survey, and the results to the multiple choice questions can therefore be considered statistically representative. The open questions in the survey were only answered by few respondents and can not be generalized in the same way.

The case studies on the other hand, studied only two municipalities, and can not so easily be generalized to all Swedish municipalities. This was of course also not the goal of the study. The goal of the explorative study was to get some understanding for the problems that municipalities are facing when trying to include their IT systems in their emergency management. Even though many of the same problems occur at both the municipalities under study, this is no proof that they appear in all Swedish municipalities. When combining the survey and the case studies, we can at least conclude that some of the problems are very common, and we can suspect that some of their causes and effects is probably the same for many more Swedish municipalities.

An important threat to the validity in this study is the possibility of researcher bias. All the interviews were conducted by the same researchers and the conclusions from the first interviews were used to steer the later ones. Because of the open form of the interviews, it would be even easier for the researchers to steer the respondents to certain conclusions. To minimise the effect of researcher bias, the interviews were conducted with two researchers present and extra care was given to let the interviewees tell their own story, without guiding their answers. In the analysis of the interviews the possibility of researcher bias was constantly taken into account when building explanations.

A threat to the construct validity that is often present when data is collected through interviews is the possibility that the participants are focusing too much on their own

side of the story and give a distorted view of reality. One reason for this is that people do not have perfect recollection, and only remember a part of what happened. A second reason is that people automatically try to defend their own actions, and although they would hopefully not lie deliberately, they might neglect to tell some things that makes them look bad. Through the use of triangulation, by interviewing different people at the same municipality and by asking different questions concerning the same topic, the effect of this can be reduced. Overall, the interviewees were not afraid at all to talk about problems they were experiencing or had experienced in the past. Because all official documents that are not declared classified are automatically public in Sweden, it was also no problem to gain access to any documents requested for analysis.

A final important threat to the validity is that the municipalities that were studied are listed as good examples of emergency management on SEMA's website: municipality A for using the MVA (Hallin et al. 2004) technique and assisting in the development of this technique, and municipality B for the risk incident reporting system they developed. This is an indication that both municipalities might be more mature in handling these issues than most other Swedish municipalities. For this exploratory study this was considered an advantage, since this allowed us to interview more experienced participants, but it makes the results harder to generalize.

7 Conclusions and Future Work

In this report we studied how municipalities in Sweden evaluate the dependability of their IT systems in possible crisis situations. A first set of case studies and the results of a survey have given us a better understanding of the main challenges involved.

In the case studies we noted a number of problem areas. The main problem is that the studied municipalities lack a forum where preventive measures concerning IT dependability issues can be discussed. All involved parties do their best in contributing to the dependability of the systems, but no cooperation to discuss these matters on a strategical level is present. Therefore, those responsibilities that lie on the border between different people's areas of responsibility are often given too little attention.

Now that we have identified a problem and possibility for improvement with how municipalities deal with dependability of their IT systems, a next logical step is to start working towards a tool that can help municipalities improve in this field. In the end this should result in a process improvement model that is simple enough to be applied even by small municipalities, but that at the same time can make a big difference. The focus of this improvement model should be in stimulating the communication about these issues between the IT personnel, the emergency managers and the users of the different IT systems in the municipality.

The first step towards this goal is to develop a measurement scale and tool that municipalities can use to assess how mature they are in handling this issue and in which areas there is most room for improvement. The next step is then to evaluate this measurement tool in practice and improve it based on this evaluation.

With the help of this measurement tool we can then start to develop a process improvement model based on these measurements that helps municipalities reach a higher level of maturity in dealing with dependability issues and to sustain these improvements. The final step would then be to evaluate this complete maturity model in a practical setting at one or more municipalities while continuously improving it based on these experiences and the feedback we receive.

Bibliography

A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1(1):11–33, 2004.

BITS. Basic Level for IT Security, 2003. Swedish Emergency Management Agency, SEMA recommends 2003:2.

COBIT. Control objectives for information and related technologies (COBIT) (3rd ed.), 2000. IT Governance Institute, USA.

K. Goševa-Popstojanova and S. Kamavaram. Software reliability estimation under uncertainty:generalization of the method of moments. In *Proceedings of the 8th IEEE International Symposium on High Assurance Systems Engineering*, pages 209–218, 2004.

P.-O. Hallin, J. Nilsson, and N. Olofsson. Kommunal sårbarhetsanalys, 2004. KBM:s forskningsserie, nr 3.

ISO-IEC 17799. ISO-IEC 17799: Information technology - Security techniques - Code of practice for information security management, 2005. International Organization for Standardization.

ITIL. Information Technology Infrastructure Library (ITIL) Version 3, 2007. Office of Government Commerce.

K. Kalmelid and J. Gustavsson. Inventering av kompetensbehov m.m. inom informationssäkerhet i offentlig sektor. Technical report, Rapport, Informationssäkerhets- och analysenheten, Krisberedskapsmyndigheten, 2005.

KBM, 2004. Kommunens plan för hantering av extraordinära händelser, 2004. Krisberedskapsmyndigheten, KBM Rekommenderar 2004:1.

KBM 2005. Samhällets krisberedskap - Inriktning för verksamheten 2007, 2005. Krisberedskapsmyndigheten, Planeringsprocessen 2005:3.

KL(1991:900). Kommunallag (1991:900), 1991. Finansdepartementet KL.

J. N. Luftman. *Managing the Information Technology Resource: Leadership in the Information Age*. Prentice-Hall, 2003.

C. Robson. *Real World Research: A Resource for Social Scientists and Practitioner-researchers*. Blackwell Publishers, second edition, 2002.

SALAR 2005. Levels of Local Democracy in Sweden, 2005. Swedish Association of Local Authorities and Regions.

SEMA, 2007. URL: `http://www.krisberedskapsmyndigheten.se/` (Last accessed November 2007). Swedish Emergency Management Agency.

K. Weyns and P. Runeson. Sensitivity of Software System Reliability to Usage Profile Changes. In *Proceedings of the 2007 ACM Symposium on Applied Computing*, pages 1440–1444, 2007.

R. K. Yin. *Case Study Research: Design and Methods*. SAGE Publications Ltd, 2003.